

PROCESSAMENTO DIGITAL DE IMAGENS PARA UM DETECTOR DE  
PARTÍCULAS DE ALTA ENERGIA E ELEVADA TAXA DE EVENTOS

Aline da Rocha Gesualdi Mello

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO  
DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA  
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS  
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR  
EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

---

Prof. José Manoel de Seixas, D.Sc.

---

Prof. Márcio Portes de Albuquerque, D.Ing.

---

Prof. Gelson Vieira Mendonça, Ph.D.

---

Prof. Érica Polycarpo, D.Sc.

---

Prof. Ricardo Tadeu Lopes, D.Sc.

---

Prof. Francisco Marcos de Assis, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 2006

MELLO, ALINE DA ROCHA GESUALDI

Processamento Digital de Imagens  
para um Detector de Partículas de Alta  
Energia e Elevada Taxa de Eventos [Rio  
de Janeiro]2006

X,153 pp 29,7 cm (COPPE/UFRJ, D.Sc.,  
Engenharia Elétrica, 2006)

Tese - Universidade Federal do Rio de  
Janeiro, COPPE

1.Processamento Digital de Imagens  
2.Transformada de Hough 3.Morfologia  
Matemática 4.Seleção Geométrica de  
Pontos 5.Detector ATLAS 6.Física de  
Partículas

I.COPPE/UFRJ      II.Título (série)

## Agradecimentos

A Deus por todas as bênçãos.

Aos meus orientadores José Manuel de Seixas e Márcio Portes de Albuquerque pelos ensinamentos recebidos no exercício dessa pesquisa. Agradeço imensamente por toda a paciência, profissionalismo e humanidade em todos os momentos.

A todos os professores do LPS, sem exceção, o apoio em diversas oportunidades.

A meus pais que sempre me conduziram para o caminho do bem e por me ensinarem a concretizar sonhos. A minha família, meus grandes incentivadores, por entender e apoiar os momentos de ausência, inquietude, ansiedade e tristeza com todo o amor e companheirismo.

Aos meus amigos do Laboratório de Processamento de Sinais e do Centro Brasileiro de Pesquisas Físicas pela amizade, profissionalismo e colaboração.

As agências de fomento CNPq e CAPES pelas bolsas de doutoramento e doutoramento sanduiche no exterior.

Ao Laboratório de de Processamento de Sinais (LPS-COPPE/UFRJ), por ter proporcionado mais esta oportunidade de evoluir em minha formação.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

PROCESSAMENTO DIGITAL DE IMAGENS PARA UM DETECTOR DE  
PARTÍCULAS DE ALTA ENERGIA E ELEVADA TAXA DE EVENTOS

Aline da Rocha Gesualdi Mello

Março/2006

Orientadores: José Manoel de Seixas

Márcio Portes de Albuquerque

Programa de Engenharia Elétrica

Este trabalho apresenta a aplicação da transformada de Hough para detecção *on-line* de vértices primários num ambiente de colisão de partículas em períodos de 25 ns. Os vértices primários são obtidos através da detecção de retas na imagem volumétrica que representa o produto das colisões. Apesar da densidade das imagens, os requisitos de operação exigem que os algoritmos de localização de vértices possuam uma latência de execução máxima de 1 ms. Estes requisitos são repetidos mesmo quando as colisões ocorrem na sua maior luminosidade, o que gera ruído adicional por empilhamento. A transformada de Hough nas formas local, adaptativa e morfológica foram aplicadas com sucesso a este problema, atingindo uma eficiência de detecção de 98,66% em  $0,0203(\pm 0,0165)$ ms nas condições emuladas de operação. A implementação usa uma seleção específica na formatação de dados, a qual acelerou significativamente o acesso às imagens nas regiões de colisão.

Abstract of the Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

DIGITAL IMAGE PROCESSING APPLIED TO A PARTICLE DETECTOR IN  
A HIGH ENERGY PHYSICS ENVIRONMENT

Aline da Rocha Gesualdi Mello

March/2006

Advisors: José Manoel de Seixas

Márcio Portes de Albuquerque

Department: Electrical Engineering

This work describes an online application of Hough transform to detect primary vertices in a particle collision environment in periods of 25 ns. The primary vertices are obtained by detecting lines in the volumetrical image produced by the debris of the collisions. Algorithms operation requirements to vertex finding have a maximum latency of 1 ms. These requirements have to be fulfilled even when collisions occurs at high luminosity, which generate additional pile-up noise. The Hough transform technique in its local, adaptive and morphological form were succesfully applied to this problem, achieving a detection efficiency of 98,66% in 0,0203( $\pm 0,0165$ )ms in system operation emulation. The implementation uses a specific data format, that accelerated significantly the access to the images at collision region.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
	<b>Introdução</b>	<b>1</b>
1.1	Delimitação e Motivação . . . . .	2
1.2	Justificativa e Objetivos . . . . .	3
1.3	Metodologia . . . . .	4
1.4	Organização do Documento . . . . .	5
<b>2</b>	<b>O Experimento ATLAS e o Sistema HLT</b>	<b>7</b>
2.1	Introdução à Física de Partículas . . . . .	8
2.2	CERN - Laboratório Europeu de Física de Partículas . . . . .	9
2.3	O Experimento ATLAS . . . . .	11
2.3.1	Sistema de Coordenadas do ATLAS . . . . .	12
2.3.2	Os Detectores de Traço . . . . .	13
2.3.3	O Sistema de Filtragem do ATLAS . . . . .	20
2.4	O Sistema HLT ( <i>High Level Trigger</i> ) . . . . .	21
2.4.1	Os Critérios de Seleção . . . . .	22
2.4.2	Objetos de Física para Seleção de Eventos . . . . .	23
2.4.3	Arquitetura do Sistema HLT . . . . .	23
2.4.4	Implementação da Arquitetura . . . . .	24
<b>3</b>	<b>Reconstrução de Vértices - Estado da Arte</b>	<b>27</b>
3.1	Sistemas de Localização de Vértices Primários . . . . .	28
3.1.1	Localização 3D por Correlação Cruzada . . . . .	28
3.1.2	Localização por Filtragem de Kalman . . . . .	31
3.1.3	Localização pelo Método de Billoir . . . . .	32

3.1.4	Localização por Filtros de Somas Gaussianas . . . . .	35
3.1.5	Localização por Filtros de <i>Annealing</i> Determinísticos . . . . .	37
3.1.6	Localização pelo Método dos Multiplicadores de Lagrange . . . . .	39
3.1.7	Localização pelo Método de Ajuste Linear ( <i>Least Square</i> ) . . . . .	41
3.2	Sistemas de Localização de Vértices Primários do ATLAS . . . . .	43
3.2.1	Localização por Ajuste Linear de Retas ( <i>ZFinder</i> ) . . . . .	43
3.2.2	Localização através do Calorímetro Eletromagnético . . . . .	45
3.3	Sistemas de Reconstrução de Trajetórias de Partículas . . . . .	49
3.3.1	Reconstrução de Trajetórias por Filtragem de Kalman . . . . .	49
3.3.2	Reconstrução de Trajetórias por Braços Elásticos . . . . .	50
3.3.3	Reconstrução de Trajetórias por Transformada de Hough . . . . .	52
<b>4</b>	<b>A Transformada de Hough</b> . . . . .	<b>55</b>
4.1	Transformada de Hough Padrão - THP . . . . .	56
4.1.1	Detecção de Retas . . . . .	56
4.1.2	Descrição de Retas por Funções Senoidais . . . . .	58
4.1.3	Discretização da Transformada de Hough Padrão . . . . .	60
4.1.4	Presença de Ruído no Espaço de Dados . . . . .	62
4.1.5	Considerações . . . . .	63
4.2	Transformada de Hough Local (THL) . . . . .	63
4.2.1	Características . . . . .	63
4.2.2	Considerações . . . . .	66
4.3	Transformada de Hough Adaptativa (THA) . . . . .	67
4.3.1	Características . . . . .	67
4.3.2	Considerações . . . . .	67
4.4	Transformada de Hough Morfológica (THM) . . . . .	70
4.4.1	Características . . . . .	71
4.4.2	Considerações . . . . .	73
<b>5</b>	<b>O Conjunto de Dados</b> . . . . .	<b>75</b>
5.1	A Geração dos Dados . . . . .	75
5.2	Caracterização do Vértice Primário de Colisão . . . . .	78
5.3	Ilustrando os Eventos . . . . .	78

5.4	Conjunto de Dados Utilizado . . . . .	81
<b>6</b>	<b>O Sistema de Seleção de Regiões</b>	<b>83</b>
6.1	Estratégia de Seleção e Princípios Gerais . . . . .	84
6.2	Implementação e Projeto do Seleccionador de Regiões . . . . .	86
6.3	Estudo da Performance do Sistema . . . . .	93
<b>7</b>	<b>Apresentação e Análise do Algoritmo de Referência <i>ZFinder</i></b>	<b>96</b>
7.1	Algoritmo para o Cálculo de $z_{vtx}$ . . . . .	97
7.2	Cálculo de $z_{vtx}$ Utilizando Trios de Pontos Tridimensionais . . . . .	100
7.3	Resultados do Algoritmo <i>ZFinder</i> em Eventos de Elétrons Individuais	101
<b>8</b>	<b>Sistema de Localização de Vértices Primários por Transformada de Hough</b>	<b>105</b>
8.1	Os Dados de Entrada . . . . .	107
8.2	Abordagem de Seleção de Pontos . . . . .	108
8.2.1	Seleção por Cone . . . . .	108
8.2.2	Seleção por Raio de Atuação em $\phi \times z$ . . . . .	110
8.2.3	Seleção por Vizinhança em $\phi$ . . . . .	111
8.3	Abordagem de Localização . . . . .	113
8.3.1	Transformada de Hough com Conhecimento Especialista . . . . .	113
8.3.2	Transformada de Hough Local - THL . . . . .	114
8.3.3	Transformada de Hough Adaptativa - THA . . . . .	116
8.3.4	Transformada de Hough Morfológica - THM . . . . .	119
8.4	Implementação dos Métodos . . . . .	123
8.5	Reconstrução de Vértices Primários no Ambiente ATHENA . . . . .	125
8.6	Reconstrução de Vértices Primários no SHARC ADSP-21160 . . . . .	127
<b>9</b>	<b>Apresentação, Análise e Interpretação dos Resultados</b>	<b>130</b>
9.1	Reconstrução de Vértices Primários para Elétrons com Empilhamento	131
9.1.1	Eficiências e Tempos de Processamento . . . . .	131
9.1.2	Comparação entre os Métodos de Reconstrução por Hough e o <i>ZFinder</i> . . . . .	135

9.2	Reconstrução de Vértices Primários no Ambiente de Desenvolvimento	
	ATHENA . . . . .	144
9.3	Reconstrução de Vértices Primários no SHARC ADSP-21160 . . . . .	146
9.4	Considerações . . . . .	149
<b>10</b>	<b>Conclusão</b>	<b>151</b>
10.1	Trabalhos Futuros . . . . .	152
	<b>Bibliografia</b>	<b>154</b>
<b>A</b>	<b>Publicações</b>	<b>162</b>

# Lista de Figuras

2.1	Complexo LHC. . . . .	11
2.2	Diagrama esquemático do detector ATLAS. . . . .	11
2.3	A interação das partículas com os diversos detectores. . . . .	12
2.4	Sistema de coordenadas cartesianas (XYZ) com Z na direção do feixe. . . . .	13
2.5	Representação da coordenada cilíndrica ( $\phi\rho Z$ ) do ATLAS, onde $\phi$ corresponde ao ângulo polar e $\rho$ ao raio do cilindro. . . . .	14
2.6	Variação da pseudo-rapidez $\eta(\theta)$ . O destaque hachurado representa a região de aceitação dos detectores do ATLAS que é de $ \eta  \leq 2,5$ . . . . .	14
2.7	Representação tridimensional dos Detectores Internos do ATLAS. . . . .	15
2.8	Secção transversal ( $\rho \times z$ ) da geometria dos Detectores Internos. . . . .	16
2.9	Esquema do sistema de filtragem do ATLAS (TDAQ). . . . .	21
2.10	Diagrama da arquitetura de objetos do ATHENA. . . . .	26
3.1	Distribuição do vértice nos eixos $x$ e $y$ . Extraído de [1]. . . . .	29
3.2	Diferença entre o valor desejado e o obtido em função da distância radial da localização do vértice com relação ao eixo $z$ . (a) $x_{true}$ e $x_{found}$ ; (b) $y_{true}$ e $y_{found}$ ; c) $z_{true}$ e $z_{found}$ . Extraído de [1]. . . . .	30
3.3	Vista $x \times y$ dos detectores de traço do ATLAS. Extraído de [2]. . . . .	33
3.4	Seqüência de rastreamento de trajetórias. $x_n^m$ representa a coordenada na $n$ -ésima camada que pertence a trajetória reconstruída. Extraído de [3]. . . . .	34
3.5	Resolução dos vértices primários reconstruídos de eventos na direção $z$ do processo $WH(120) \rightarrow \mu\nu u\bar{u}$ sem empilhamento. Extraído de [3]. . . . .	35

3.6	Resolução dos vértices primários reconstruídos de eventos na direção $z$ do processo $WH(130) \rightarrow lll$ com empilhamento e a alta luminosidade. O vértice primário correto foi encontrado em 88% dos casos. Extraído de [3]. . . . .	36
3.7	Cobertura de 50% (esquerda), 90% (meio) e eficiência relativa (direita) dos ajustes dos vértices para trajetórias com duas componentes. Extraído de [4]. . . . .	37
3.8	Ilustração do processo de recursão determinística (DA) para o aglomeramento de um conjunto de pontos. Extraído de [5]. . . . .	38
3.9	Ilustração do algoritmo de ajuste linear. Extraído de [6]. . . . .	41
3.10	Correlação entre as coordenadas $Z_A^i$ e $Z_B^i$ para trajetórias oriundas do vértice primário. (a) Medida de todas as combinações das coordenadas em $A$ e $B$ . Note que a escala no plano $B$ é duas vezes maior que no plano $A$ . As combinações verdadeiras formam uma linha reta. (b) Contagem da combinação de todas as diagonais. (c) somente as combinações verdadeiras são contadas. (d) Contagem das combinações erradas. Extraído de [6]. . . . .	42
3.11	Exemplo de uma região de interesse do LVL2 no plano $\rho \times z$ . . . . .	46
3.12	Região de interesse proposta no plano $\rho \times z$ . . . . .	46
3.13	Determinação dos valores das Equações 3.19 e 3.20. . . . .	47
3.14	Representação dos parâmetros da trajetória nos eixos $x$ e $y$ . . . . .	50
3.15	Ilustração do processo de distribuição gaussiana nos braços (ou trajetórias). . . . .	51
3.16	Resultado do algoritmo de braços elásticos em uma simulação de evento. . . . .	52
3.17	Exemplo de um evento de colisão. (a) imagem $x \times y$ do detector de traço SFT. (b) simulação da imagem em (a). . . . .	53
3.18	Exemplo de um resultado da transformada de Hough em um evento de colisão do SFT. . . . .	54
4.1	Exemplo de aplicação da THP em um conjunto de quatro pontos que definem uma reta de coeficiente angular igual a -1 e linear igual a 8. . . . .	57

4.2	Exemplo do espaço $m, c$ . Os quatro pontos do plano- $xy$ tornaram-se retas no plano- $mc$ . O ponto de interseção indica o coeficiente angular e linear da reta no plano- $xy$ . . . . .	58
4.3	Representação de $\rho$ e $\theta$ . . . . .	58
4.4	Exemplo de aplicação da THP em um conjunto de quatro pontos que definem uma reta com $\rho = \sqrt{32} \approx 5,657$ e $\theta = \pi/4 \approx 0,7854$ . . . . .	59
4.5	Exemplo do espaço $\rho, \theta$ . Os quatro pontos do plano- $xy$ tornaram-se funções senoidais no plano- $\rho\theta$ . O ponto de interseção indica o coeficiente angular e linear da reta no plano- $xy$ . . . . .	60
4.6	Seleção de um par de pontos para o cálculo da THL. . . . .	65
4.7	Fluxograma para a THL. . . . .	65
4.8	Fluxograma para o algoritmo da THA. . . . .	68
4.9	Ilustração das máscaras utilizadas na THM. . . . .	73
4.10	Fluxograma para o algoritmo da THM. . . . .	74
5.1	Esquema do processo de simulação de dados no ATLAS. . . . .	76
5.2	Esquema tridimensional do detector PIXEL. . . . .	79
5.3	Primeiro barril do detector PIXEL, com 4cm de raio. Os elementos sobrepostos são os módulos. . . . .	79
5.4	Três vistas do evento de elétron individual com baixa luminosidade nos detectores de traço PIXEL e SCT. (a) $\phi \times \rho$ . (b) $\rho \times z$ e (c) $\phi \times z$ . 80	
5.5	Exemplo de um elétron (a) com baixa luminosidade e (b) com alta luminosidade observados nos detectores de traço PIXEL e SCT. $z$ é a direção do feixe e $\rho$ é o raio das camadas dos detectores de traço. . . . .	82
6.1	Modelo de acesso aos dados do PESA, referenciado como esquema Londres ( <i>London scheme</i> ), pelo qual os algoritmos do HLT solicitam dados de eventos. Uma solicitação de um determinado algoritmo do HLT ao selecionador de regiões consiste na primeira etapa do sistema de nível 2[7]. . . . .	85
6.2	Exemplos de regiões . . . . .	87
6.3	Ilustração do diagrama de seqüência conceitual do Seleccionador de Regiões . . . . .	88

6.4	Ilustração do <i>EtaPhiMap</i> . Cada elemento pode conter identificadores repetidos com relação a seus vizinhos, devido a sobreposição dos componentes dos detectores. . . . .	90
6.5	Diagrama simplificado de classes do Seleccionador de Regiões. Duas classes diferentes <i>Data</i> e <i>Element</i> formam os detectores, que por sua vez pode ter sobreposição <i>RegSEelect</i> . . . ou não ter <i>LArRegSelect</i> . . . . . Explicação mais detalhada encontra-se no texto. . . . .	91
6.6	Tempos de execução para o subdetector Pixel . . . . .	95
7.1	Histograma- $z$ para um evento de baixa luminosidade e alto $p_T$ , sem empilhamento. . . . .	99
7.2	Histograma- $z$ para um evento de alta luminosidade e alto $p_T$ , sem empilhamento. . . . .	99
7.3	Histograma- $z$ para um evento de alta luminosidade e alto $p_T$ , utilizando trio de pontos para formar o histograma. . . . .	101
7.4	Tempos de execução do <i>ZFinder</i> para eventos de alta luminosidade. (a) histograma do tempo de execução; (b) gráfico do tempo de execução com relação ao número de pontos nos eventos. . . . .	102
7.5	Tempos de execução do <i>ZFinder</i> para eventos de baixa luminosidade. (a) histograma do tempo de execução; (b) gráfico do tempo de execução com relação ao número de pontos nos eventos. . . . .	103
7.6	Diferença entre os valores de simulação ( $z_{true}$ ) e os valores obtidos pelo <i>ZFinder</i> ( $z_{vtx}$ ) para amostras de baixa (a) e alta (b) luminosidade. . . . .	104
8.1	Diagrama de blocos simplificado do sistema de reconstrução de vértices primários por transformada de Hough. . . . .	106
8.2	Exemplo do método de seleção por Cone aplicado a um evento de alta luminosidade. (a) projeção $(\rho, z)$ do evento; (b) projeção $(\phi, z)$ ; (c) projeção $(\rho, z)$ após aplicação da seleção; (d) projeção $(\phi, z)$ após aplicação da seleção. . . . .	109
8.3	Exemplo do método de seleção por raio de atuação em $\phi \times z$ . (a) evento original; (b) ilustração do cilindro de atuação $(\phi, z)$ ; (c) evento filtrado; (d) projeção $(\phi, z)$ após aplicação da seleção. . . . .	110

8.4	Exemplo do método de seleção por vizinhança $\phi$ aplicado a um evento de alta luminosidade. (a) evento original; (b) mapa bidimensional $(\phi, z)$ ; (c) pontos selecionados; (d) evento filtrado. . . . .	112
8.5	Exemplo de utilização da combinação de pontos na localização do vértice primário. (a) espaço discreto $z \times \phi$ do evento selecionado pelo método de cone; (b) critério de $\rho$ distintos; (c) critério de $\rho$ vizinhos; (d) critério do centro da região selecionada para as bordas. . . . .	115
8.6	Diagrama de blocos da transformada de Hough local. . . . .	117
8.7	Diagrama de blocos da transformada de Hough adaptativa. . . . .	120
8.8	Diagrama de Venn das transformadas de Hough local e adaptativa. . . . .	121
8.9	Ilustração das máscaras utilizadas na THM. (a) até (e) ilustram as máscaras utilizadas nos espaços de parâmetro $5 \times 10$ ; (f) máscara utilizada no espaço de parâmetros $20 \times 100$ . . . . .	122
8.10	Diagrama de blocos da transformada de Hough morfológica. . . . .	124
8.11	Diagrama de Venn das três transformadas. . . . .	125
8.12	Diagrama de blocos do sistema de reconstrução de vértices primários. . . . .	126
9.1	Eficiência média e RMS dos métodos de localização de vértices (RD, RV e CB) de acordo com os métodos de seleção de pontos (CN, FZ, VZ). . . . .	133
9.2	Tempo médio de execução dos métodos de localização de vértices (RD, RV e CB) de acordo com os métodos de seleção de pontos (CN, FZ, VZ). . . . .	133
9.3	Tempo de execução do sistema de seleção geométrica de pontos, em milissegundo. Eventos de alta luminosidade. . . . .	139
9.4	Tempo de execução do sistema de seleção geométrica de pontos, em milissegundo. Eventos de baixa luminosidade. . . . .	140
9.5	Tempo de processamento com relação ao número de pontos por evento em alta luminosidade. . . . .	141
9.6	Tempo de processamento com relação ao número de pontos por evento em baixa luminosidade. . . . .	142

9.7	Tempo de execução do sistema de localização para a configuração de escolha de pontos por $\rho$ distintos e número de recursão fixa. Eventos de alta luminosidade. . . . .	143
9.8	Tempo de execução do sistema de localização para a configuração de escolha de pontos por $\rho$ distintos e número de recursão fixa. Eventos de baixa luminosidade. . . . .	144
9.9	Diferença entre os valores de simulação ( $z_{true}$ ) e os valores obtidos pelo VZCBTHMFX ( $z_{vtx}$ ) para amostras de baixa (a) e alta (b) luminosidade. . . . .	145
9.10	Tempo em ms de execução no ADSP-BF533SKBC750. (a) recursão 0; (b) recursão 1; (c) recursão 2; (d) recursão 3; (e) tempo total. . . .	148
9.11	Gráfico do tempo de execução com relação ao número de pontos nos eventos. . . . .	149

# Lista de Tabelas

2.1	Gerações de partículas de matéria. . . . .	9
2.2	Parâmetros do Detector Interno - Valores típicos de posição, área, resolução e cobertura de atuação dos detectores. . . . .	18
3.1	Eficiência do algoritmo <i>ZFinder</i> em eventos de elétrons individuais com baixa e alta luminosidade para regiões de interesse reduzidas pelo método de localização através da informação do calorímetro e para regiões de interesse originais obtidas do LVL1. . . . .	48
3.2	Tempos de processamento do algoritmo <i>ZFinder</i> em eventos de elétrons individuais com baixa e alta luminosidade para regiões de interesse reduzidas pelo método de localização através da informação do calorímetro e para regiões de interesse originais obtidas do LVL1. . . . .	48
3.3	Eficiências obtidas para os diversos algoritmos testados neste trabalho. . . . .	54
5.1	Características dos dados utilizados. . . . .	81
6.1	Ilustração do diagrama de seqüência conceitual do Seleccionador de Regiões	89
6.2	Variações do método <code>RegionSelector</code> . . . . .	92
6.3	Correspondências entre métodos e detectores. . . . .	93
6.4	Representação dos tempos médios de execução do algoritmo Seleccionador de Regiões . . . . .	94
9.1	Legenda das siglas utilizadas nas Tabelas 9.2, 9.3 e 9.4. . . . .	135
9.2	Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção de Cone. Os valores entre parênteses correspondem ao valor RMS medido. . . . .	136

9.3	Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção por raio de atuação em $\phi \times z$ . Os valores entre parênteses correspondem ao valor RMS medido. . . . .	137
9.4	Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção por vizinhança. Os valores entre parênteses correspondem ao valor RMS medido. . . . .	138
9.5	Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários testadas no ambiente ATHENA. Os valores entre parênteses correspondem ao valor RMS medido. . . .	147

# Capítulo 1

## Introdução

A análise quantitativa e a interpretação de imagens constituem ferramentas importantes em diversas áreas científicas e tecnológicas. Estas ferramentas possibilitam, por exemplo, a automação de sistemas de segurança, de mapas cartográficos e de sistemas de visão para robôs.

Atualmente, a área de melhoria de imagens continua em expansão. As aplicações são variadas podendo utilizar imagens estáticas ou dinâmicas. Normalmente, elas utilizam técnicas de melhoria ou recuperação de imagens aliadas com métodos de reconhecimento de padrões. A dificuldade destas técnicas reside no elevado volume de dados, o que pode ser dramático quando há necessidade de processamento *online* ou em tempo real. Nestes casos mais complexos, eventualmente, a solução requer a utilização de ambientes de processamento distribuído.

O ambiente de física de altas energias constitui um complexo exemplo de aplicação do processamento *online* de imagens provenientes dos detectores que rastreiam as partículas com que eles interagem. Isto pode ser observado pela utilização de um conjunto de detectores com características diferentes e inúmeros canais de detecção responsáveis pelo rastreamento das partículas. Deste modo, a imagem volumétrica dos rastros das partículas, formada por este conjunto de detectores, é bastante densa. Normalmente, a informação principal encontra-se mascarada por uma grande quantidade de ruído. Desta forma, nesse ambiente, desenvolvem-se sofisticados sistemas de filtragem que requerem processamento de elevados volumes de dados em curtos intervalos de tempo. Estes sistemas de filtragem são realizados em um ambiente de processamento distribuído, possuindo um menu de filtragem

com inúmeras possibilidades de utilização, onde diversos testes com diferentes configurações são realizados, objetivando encontrar as mais importantes. O desafio do processamento de imagens consiste em filtrar as informações relevantes contidas nas imagens volumétricas dos detectores, utilizando um ambiente de processamento distribuído e operando de forma *online*.

## 1.1 Delimitação e Motivação

Essa tese aborda a construção de um sistema *online* para o processamento de imagens de colisões de partículas para o acelerador LHC<sup>1</sup> do CERN<sup>2</sup>, a partir da informação produzida pelo detector ATLAS<sup>3</sup>. Um evento de colisão é representado por uma imagem volumétrica de 7 camadas, com diferentes granularidades por camada. O desafio consiste em utilizar o conceito de seleção de regiões e a transformada de Hough para extrapolar a informação de trajetória contida nestas imagens, a fim de localizar o ponto de colisão, ou vértice primário. O foco do estudo consiste na localização de vértices primários de elétrons com elevada acurácia e mínimo tempo de execução.

O anel colisionador LHC faz o ATLAS operar a uma taxa elevada, com cerca de 40 milhões de eventos por segundo. Estima-se que o volume de dados de uma colisão seja de 160 Gbytes/s, onde menos de 1% deste volume constitui informação relevante para o estudo da física de partículas. O desafio do sistema de filtragem do ATLAS consiste em detectar a informação relevante de forma *online*. De modo a operar com estas taxas, o sistema de filtragem e aquisição de dados do ATLAS, o sistema TDAQ (do inglês, *Trigger and Data Acquisition*), possui três estágios de execução. Estes estágios são executados em cascata, com complexidade e tempo de processamento crescente.

---

<sup>1</sup>*Large Hadron Collider*

<sup>2</sup>Centro Europeu de Física de Partículas

<sup>3</sup>*A Toroidal LHC Apparatus*

## 1.2 Justificativa e Objetivos

A construção do detector ATLAS está sendo realizada em um sistema de cooperação internacional entre especialistas de diferentes áreas. O trabalho aqui proposto está sendo desenvolvido em colaboração com o grupo PESA<sup>4</sup> do ATLAS. Os enormes desafios da construção do ATLAS encontram-se não somente no desenvolvimento de tecnologias para captura e armazenamento de informações relevantes, como também na operação em condições de elevada luminosidade e curtos intervalos entre colisões, superando os requisitos históricos dos experimentos na área.

A complexidade em observar a física de interesse utilizando um sistema de detecção complexo, com inúmeros canais, operando a uma elevada taxa de colisão, faz com que o ATLAS possua grande visibilidade científica e tecnológica. O processamento *online* em alta taxa de eventos utiliza um processamento distribuído, necessitando a integração dos aspectos de implementação, algoritmos e ambientes de emulação com a física, representando um complexo desafio. Extrair informações relevantes das imagens densas do ATLAS constitui um exemplo crucial de utilização de métodos de processamento de imagens.

O acelerador de partículas LHC (*Large Hadron Collider*) do CERN irá operar com 14 TeV de energia no centro de massa e com luminosidades de até  $2 \cdot 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ , valores duas vezes superiores aos encontrados nos aceleradores atuais. Por este motivo, é prevista uma maior ocorrência de empilhamento de eventos. O termo empilhamento (*pile-up*) significa a sobreposição de regiões sensibilizadas pela passagem de partículas. O aumento na probabilidade de colisão pela utilização de uma luminosidade elevada causa o empilhamento de regiões. Desta forma, os eventos com empilhamento possuem mais ruído de fundo devido à sobreposição de partículas provenientes de diferentes vértices primários. Os sistemas de localização de vértices de colisão estão sendo ajustados para operar nesta nova faixa de luminosidade.

O objetivo desse trabalho consiste em desenvolver ferramentas e algoritmos que permitam trabalhar com as partes sensibilizadas dos detectores, chamadas regiões de interesse, identificando os canais sensibilizados e encontrar o ponto de

---

<sup>4</sup>*Physics and Event Selection Architecture*

colisão através da extrapolação tridimensional destes canais. A ferramenta de identificação dos canais sensibilizados, denominada selecionador de regiões, foi desenvolvida como motivação para o cálculo do ponto de colisão, ou vértice primário. No entanto, esta pode ser aplicada em qualquer algoritmo que necessite trabalhar apenas com as regiões de interesse dos detectores.

O trabalho desenvolvido aplica técnicas rápidas de processamento de imagens para a localização do vértice primário de regiões de interesse, em eventos de elétrons com empilhamento. A contribuição consiste no desenvolvimento das técnicas dentro das restrições de processamento do sistema de emulação de filtragem *online* do ATLAS, denominado ATHENA. Esse trabalho é viabilizado pela ferramenta de determinação de regiões de interesse, que permite a análise separada de partes dos detectores.

### 1.3 Metodologia

A integração utilizada para abordar o problema foi dividida em três partes principais: a obtenção dos dados de entrada; a análise do algoritmo de localização *ZFinder*, utilizado como referência por possuir o mesmo objetivo em localizar vértices primários; e o desenvolvimento de um algoritmo para localização do vértice de colisão através da transformada de Hough.

Para o desenvolvimento dos algoritmos de seleção, o sistema TDAQ lança mão de um complexo pacote de dados associados de simulações de colisões e das interações dos subprodutos gerados com os diferentes detectores. Para este trabalho, se utilizam trajetórias de elétrons. Entretanto, a filtragem da massa bruta de dados que alimenta o sistema de filtragem é realizada em três estágios, sendo que o trabalho aqui proposto encontra-se inserido no sistema de filtragem do ATLAS. Por este motivo, é necessário realizar a execução dos programas responsáveis pela simulação das respostas dos filtros do primeiro estágio. Um fator complicador está relacionado com o constante desenvolvimento destes programas. Neste contexto, a compreensão do funcionamento destes programas torna-se essencial para que os métodos desenvolvidos na tese possam ser testados e avaliados. Adicionalmente, como a eficiência de rejeição do primeiro nível é da ordem de 1000 para cada evento

de simulação do segundo nível, é necessário gerar 1000 eventos para que um seja de interesse, o que torna o processo de simulação computacionalmente oneroso.

O sistema de filtragem de segundo nível utiliza o conceito de regiões de interesse para um processamento seletivo dos sinais produzidos numa colisão. Os dados de entrada dos algoritmos de obtenção do vértice de colisão e de reconstrução de trajetórias são obtidos através de um processo de seleção destas regiões apontadas pelo primeiro estágio de filtragem. A etapa de seleção de regiões desenvolvida nessa tese viabiliza o processo de filtragem do segundo nível. Esta etapa foi realizada de acordo com as especificações do sistema TDAQ, encontra-se em funcionamento no ambiente de desenvolvimento ATHENA e atualmente é utilizada pelos grupos do HLT e do sistema de reconstrução de eventos.

Uma vez obtidos os dados de entrada, inicia-se a etapa de análise do algoritmo de localização *ZFinder*. Este algoritmo vem sendo utilizado pelo segundo nível de filtragem do ATLAS há aproximadamente uma década, sendo considerado como referência na análise dos resultados. A análise do algoritmo *ZFinder*, visa compreender o funcionamento deste algoritmo com o objetivo de aproveitar o conhecimento especialista utilizado. O estudo dos casos das falhas na localização do vértice de colisão também será realizado. O tempo e a eficiência de execução do algoritmo, para os diversos dados de colisão, também são verificados.

Na fase final, são propostas técnicas de processamento para a localização do vértice de colisão baseadas na transformada de Hough, contemplando os casos de eventos de elétrons com empilhamento. São realizadas comparações entre os métodos de localização por Hough com o *ZFinder* e é feita a integração com o ambiente de desenvolvimento Athena.

## 1.4 Organização do Documento

O documento encontra-se dividido em quatro grandes módulos: conceitos teóricos, desenvolvimento, resultados e conclusões.

O módulo de apresentação dos conceitos teóricos encontra-se dividido em três capítulos: o experimento ATLAS e o sistema HLT, Capítulo 2; o estado da arte dos métodos de reconstrução de vértices primários, Capítulo 3; e a transformada de

Hough, Capítulo 4.

O módulo de desenvolvimento apresenta o conjunto de dados utilizado, Capítulo 5; o sistema de seleção de regiões, Capítulo 6; é feita uma descrição detalhada do algoritmo de referência *ZFinder*, Capítulo 7; e o sistema de reconstrução de vértices primários por transformada de Hough, Capítulo 8.

O módulo de resultados do algoritmo de localização, Capítulo 9, apresenta, analisa e interpreta os resultados do sistema de reconstrução de vértices primários por transformada de Hough. São feitas comparações com o método de localização *ZFinder*. Os resultados obtidos no sistema de selecionamento de regiões estão apresentados no Capítulo 6.

Por fim, o último módulo, Capítulo 10, apresenta o desfecho do trabalho mostrando as vantagens de utilização do sistema de reconstrução de vértices primários por transformada de Hough e as perspectivas de trabalhos futuros.

## Capítulo 2

# O Experimento ATLAS e o Sistema HLT

A Física de Altas Energias busca encontrar respostas a algumas perguntas, tais como: “quais são os blocos fundamentais que constituem a matéria?” e “como as partículas interagem?”. A visão atual da estrutura da matéria é formalizada no chamado Modelo Padrão (MP). Formulado a partir do final da década de sessenta, este modelo sintetiza o conhecimento alcançado em pouco mais de um século de pesquisas em física de partículas[8].

A técnica fundamental na física de altas energias consiste em bombardear um alvo com um feixe de partículas. Os produtos da reação entre alvo e partículas são detectados. A reconstituição do desenvolvimento temporal desta reação fornece informações sobre o processo fundamental ocorrido[9].

Os últimos trinta anos da história da física de altas energias são marcados por grandes experimentos: colaborações internacionais que envolvem inúmeros profissionais de diversas áreas de conhecimento. O CERN, na Suíça, representa o maior laboratório no mundo onde são realizados experimentos de colisão de partículas[10]. Uma a uma, as peças do MP foram se encaixando, à medida que as partículas previstas pelo modelo foram sendo descobertas. A palavra *descoberta* passa a ter um sentido novo, pois, agora os experimentos são projetados para encontrar algo previsto teoricamente e fenomenologicamente avaliado como de viável detecção.

## 2.1 Introdução à Física de Partículas

O princípio de detecção de estruturas atômicas e subatômicas consiste na proposição elementar de que todas as partículas têm propriedades de onda. Desta forma, quanto menor a partícula que se deseja observar, quanto menor deve ser o comprimento de onda incidente.

No começo do século passado, os experimentos em física de partículas eram essenciais para testar e refinar as hipóteses do MP. Os detectores mais comuns eram: a câmara de Wilson e emulsões fotográficas[11]. A energia e o momento da partícula podiam ser determinados, respectivamente, através das medidas de comprimento e de curvatura dos traços.

A evolução nos estudos do MP levou à necessidade de caracterização de partículas cada vez menores. O princípio utilizado para diminuir o comprimento de onda de partículas sonda consiste em aumentar seu momento. Isto se deve à Equação de De Broglie,  $p = h/\lambda$ , que relaciona o momento ( $p$ ) com o inverso do comprimento de onda ( $\lambda$ ), onde  $h$  é a constante de Planck[12]. De forma a aumentar o momento são, portanto, utilizados aceleradores de partículas.

Atualmente, o MP é composto por seis quarks, três léptons com seus neutrinos associados e partículas mediadoras como, por exemplo, os fótons. Os quarks formam os hádrons, que se dividem em bárions e mésons. Para todos os quarks e léptons existem antipartículas correspondentes que possuem cargas opostas. As teorias atuais de interações entre partículas descrevem diferentes forças entre elas, onde a troca é realizada por partículas mediadoras ou transportadoras de força[13]. A Tabela 2.1 apresenta a geração da matéria.

Pode-se notar, na Tabela 2.1, que há três grupos distintos para os quarks e léptons, chamados de Gerações de Partículas de Matéria. Toda matéria visível no universo é feita da primeira geração de partículas que são: quarks up, quarks down e elétrons. Todas as partículas da segunda e terceira gerações são instáveis e decaem, tornando-se partículas da primeira geração, a única estável.

As partículas se relacionam através das interações fundamentais, que são: eletromagnética, forte, fraca e gravitacional. Para entender como as partículas interagem, deve-se estudar a característica das partículas mediadoras e as forças com que estão associadas.

Quarks	<b>u</b> up	<b>c</b> charmoso	<b>t</b> top
	<b>d</b> down	<b>s</b> estranho	<b>b</b> bottom
Léptons	$\nu_e$ e-Neutrino	$\nu_\mu$ $\mu$ -Neutrino	$\nu_\tau$ $\tau$ -Neutrino
	<b>e</b> elétron	$\mu$ múon	$\tau$ tau
<b>I      II      III</b>			
A geração da matéria			

Tabela 2.1: Gerações de partículas de matéria.

Mesmo que o MP faça uma boa descrição dos fenômenos observados, experimentalmente e teoricamente ainda encontra-se incompleto. Isto se deve, dentre outros aspectos, ao fato do MP não possuir um modelo matemático que estime a massa das partículas. Os físicos teorizaram a existência do bóson de Higgs, que é capaz de determinar se a partícula possui massa, mas não estima seu valor. Essa partícula, supostamente, interage com as outras de forma que as massas sejam geradas de maneira apropriada[14]. O bóson de Higgs ainda não foi observado, mas os físicos esperam provar ou descartar a sua existência em futuros estudos nos colisionadores de altas energias.

Um dos objetivos dos físicos consiste em juntar teorias aparentemente não relacionadas em uma única e simples teoria unificada (GUT, *Grand Unified Theory*). A vantagem de uma teoria única sobre muitas teorias fragmentadas é que ela, freqüentemente, oferece uma explicação mais elegante dos dados e é capaz de mostrar a direção para futuras áreas de estudo.

## 2.2 CERN - Laboratório Europeu de Física de Partículas

O CERN representa atualmente o maior laboratório do mundo em pesquisas na área de física de partículas. Possui sua sede perto de Genebra, Suíça. Desde a sua

fundação, em 1954, tem sido um exemplo bem sucedido de colaboração internacional, juntando cerca de 6500 cientistas, de mais de 80 nacionalidades.

O objetivo do CERN é a investigação científica pura, visando responder questões como: de que é constituído o nosso Universo?; de onde vem a matéria?; e como é que as partículas elementares interagem?

Para procurar responder a estas questões, o CERN utiliza aceleradores, colisionadores e detectores de partículas, dispositivos que se encontram entre os maiores e mais complexos instrumentos científicos jamais construídos. Por exemplo, o anel acelerador de partículas do LHC (*Large Hadron Collider*), experimento ora em construção no CERN, tem uma circunferência de 27 km, e alguns dos detectores que estão em construção têm as dimensões de edifícios de quatro andares[15].

O laboratório desempenha um papel fundamental no desenvolvimento de tecnologia de ponta, com vista a futuras aplicações: desde a ciência de materiais até a engenharia eletrônica ou a computação. A física das partículas requer o melhor desempenho em todos os campos e exige uma superação permanente da barreira do que é possível fazer em termos tecnológicos.

Atualmente, o CERN está construindo um novo acelerador de hádrons, o LHC, que entrará em funcionamento em 2007. Este acelerador permitirá sondar ainda mais profundamente a matéria e responder a questões ainda por esclarecer neste domínio.

O LHC é um acelerador de partículas que está sendo projetado para colidir feixes de prótons com energia de até 14TeV. Quatro grandes detectores estão sendo construídos ao redor deste acelerador: *Large Hadron Collider Beauty Experiment* (LHCb), *Compact Muon Solenoid* (CMS), *A Large Ion Collider Experiment* (ALICE) e *A Toroidal LHC Apparatus* (ATLAS).

A colaboração LHCb está construindo um detector para obter medidas precisas do decaimento de mésons  $B$ . O detector ATLAS, por sua vez, possui o objetivo de descobrir e estudar o bóson de Higgs e possíveis extensões do Modelo Padrão. A colaboração CMS está construindo um sistema de detecção de múons altamente eficiente com o mesmo fim. Por fim, o experimento ALICE está construindo um detector de íons pesados, dedicado ao estudo das interações fortes da matéria em densidades de energia elevadas. A Figura 2.1 representa o complexo LHC[16].

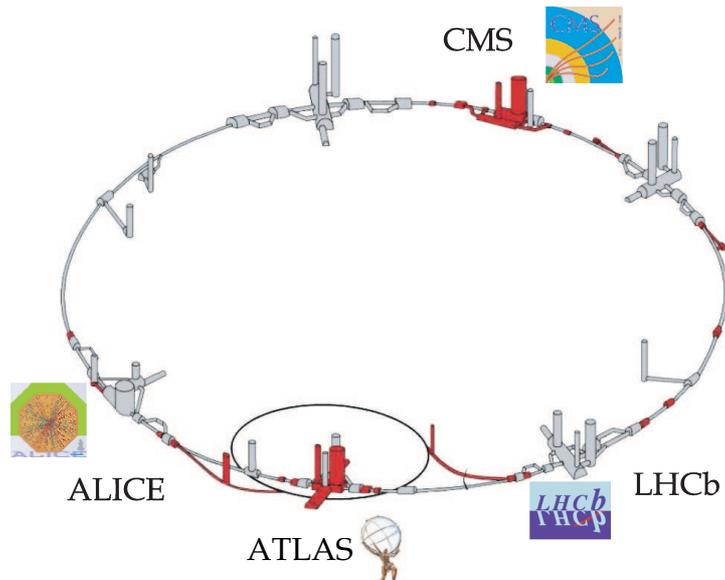


Figura 2.1: Complexo LHC.

## 2.3 O Experimento ATLAS

O experimento ATLAS será formado por quatro sistemas de detecção independentes, posicionados cilíndricamente ao redor da colisão. No interior, três detectores de traço estão sendo colocados. A seguir, virão os calorímetros eletromagnético e hadrônico e, por fim, os detectores de múons. A Figura 2.2 mostra o esquema do detector ATLAS[17]. No canto inferior esquerdo, é possível notar o desenho de pessoas em escala. O ATLAS terá altura de um edifício de quatro andares.

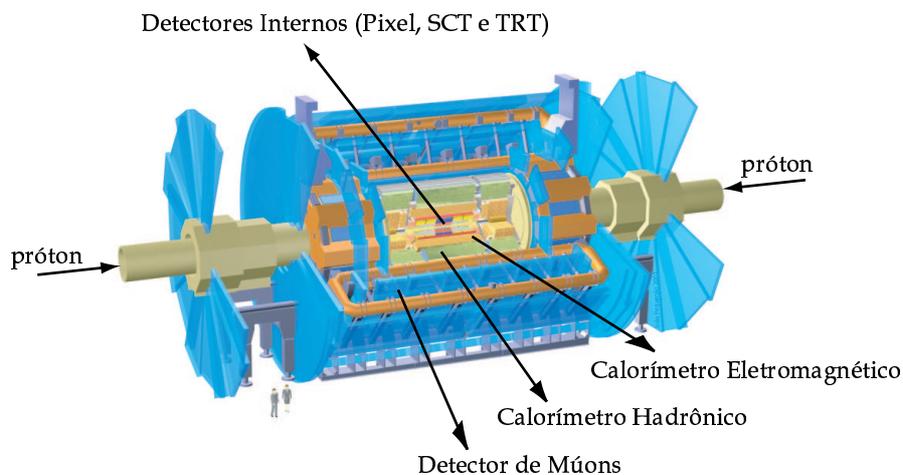


Figura 2.2: Diagrama esquemático do detector ATLAS.

A razão pela qual o ATLAS é dividido em muitos detectores é que cada um dos detectores testa conjuntos específicos de propriedades das reações produzidas pelas colisões. Esses detectores são empilhados de modo que todas as partículas atravessem as camadas diferentes sequencialmente. Uma partícula não será evidente até que interaja com os detectores de forma mensurável, ou decaia em partículas detectáveis. A Figura 2.3 apresenta uma ilustração da interação das partículas com os diversos detectores[18]. Cada partícula possui sua própria “assinatura” ao atravessar os detectores.

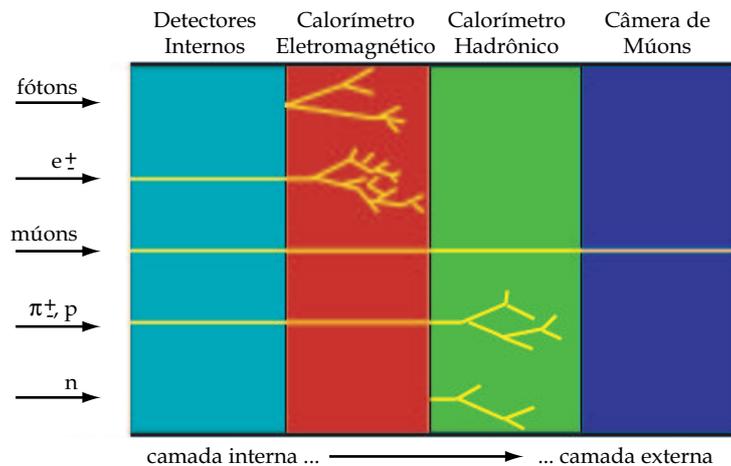


Figura 2.3: A interação das partículas com os diversos detectores.

### 2.3.1 Sistema de Coordenadas do ATLAS

O ATLAS utiliza o denominado sistema de coordenadas da mão direita, com o eixo  $x$  apontando para o centro da circunferência formada pelo túnel LHC e o eixo  $z$  é ao longo do túnel. A Figura 2.4 ilustra os eixos de coordenadas cartesianas do ATLAS[19].

Outro sistema de coordenadas bastante utilizado no ATLAS é o de coordenadas cilíndricas, como ilustra a Figura 2.5. O eixo da abscissa representa a coordenada do feixe ou  $z$ , o eixo da ordenada representa os raios ( $\rho$ ) dos cilindros e, por fim, o ângulo azimutal  $\phi$  é a terceira coordenada deste sistema. Observa-se também na Figura 2.5, o ângulo polar  $\theta$  e o momento transverso  $p_t$ . Estes dois parâmetros são bastante utilizados no estudo da física, uma vez que o ângulo polar  $\theta$  fornece

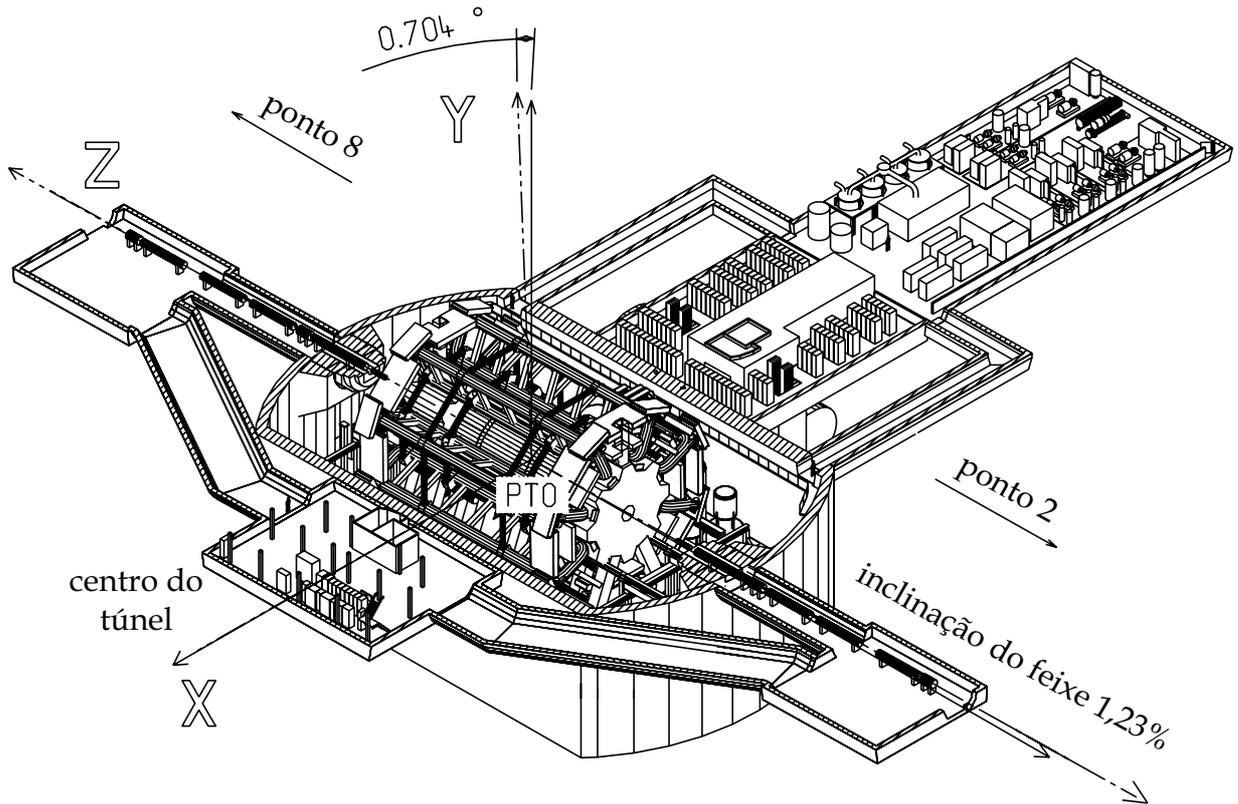


Figura 2.4: Sistema de coordenadas cartesianas (XYZ) com Z na direção do feixe.

a pseudo-rapidez (*pseudorapidity*<sup>1</sup>) da partícula e  $p_t$  seu momento perpendicular ao eixo do feixe do LHC.

A pseudo-rapidez é uma variável que aproxima a rapidez (*rapidity*) da partícula se sua massa e seu momento não forem conhecidos. Ela é simbolizada pela letra grega  $\eta$ . A Equação (2.1) apresenta o cálculo desta variável. A Figura 2.6 ilustra a variação da pseudo-rapidez em relação ao ângulo polar  $\theta$  na direção do eixo  $z$  positivo.

$$\eta(\theta) = -\log\left(\tan\frac{\theta}{2}\right) \quad (2.1)$$

Espera-se que a posição das interações próton-próton no ATLAS varie por uma faixa de  $\pm 15$  cm ao redor do centro do detector ( $z = 0$ ), ao longo da direção  $z$ . O conhecimento preciso ( $< 1$  mm) da posição de colisão ( $z_{vtx}$ ) é de interesse para diversos estudos de Física[2].

<sup>1</sup>Variável frequentemente utilizada para descrever o comportamento de partículas.

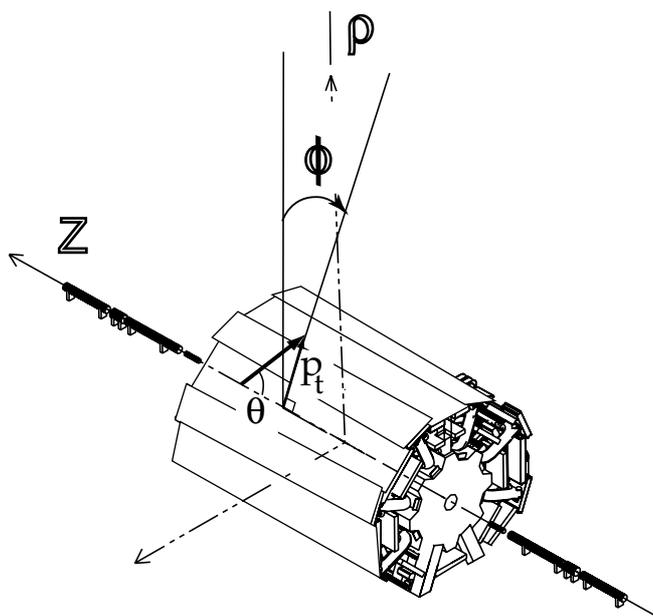


Figura 2.5: Representação da coordenada cilíndrica ( $\phi\rho Z$ ) do ATLAS, onde  $\phi$  corresponde ao ângulo polar e  $\rho$  ao raio do cilindro.

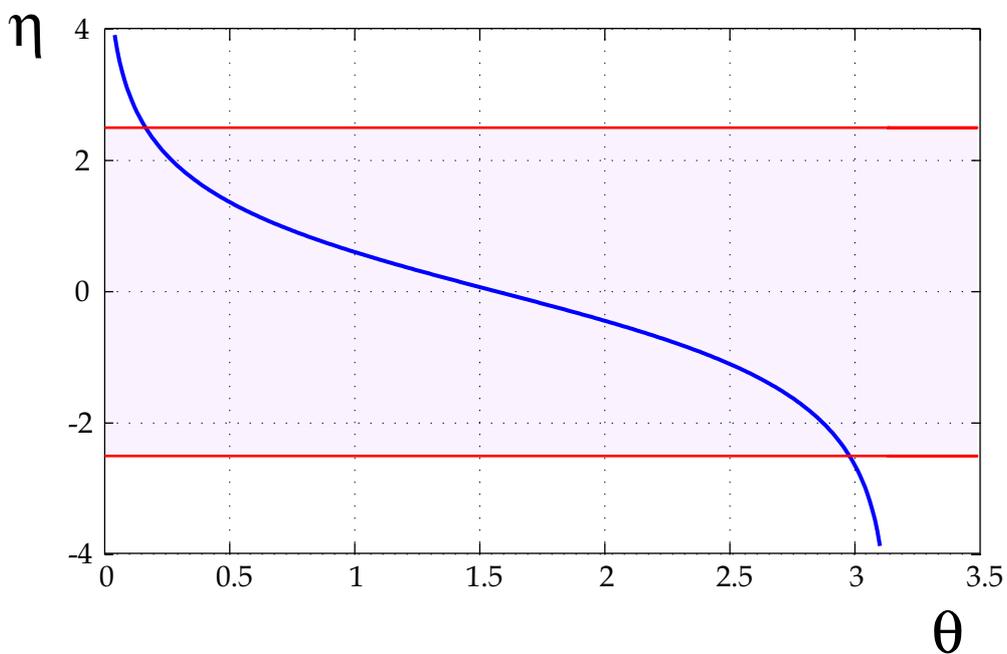


Figura 2.6: Variação da pseudo-rapidez  $\eta(\theta)$ . O destaque hachurado representa a região de aceitação dos detectores do ATLAS que é de  $|\eta| \leq 2,5$ .

## 2.3.2 Os Detectores de Traço

Os detectores internos medem as direções, momento e cargas de partículas eletricamente carregadas, produzidas em cada colisão próton-próton. Os detectores internos do ATLAS são compostos por três sistemas diferentes de sensores, todos imersos em um campo magnético ( $\approx 8\text{T}$ ) paralelo ao eixo do feixe[2].

### 2.3.2.1 Geometria dos Detectores de Traço

Os detectores internos combinam sistemas de alta resolução, nos raios mais internos, com sistemas de rastreamento contínuo de elementos nos raios mais externos. A Figura 2.7 apresenta uma secção da vista geométrica tridimensional dos detectores internos[2] e a Figura 2.8 um corte da secção transversal de um quarto dos detectores.

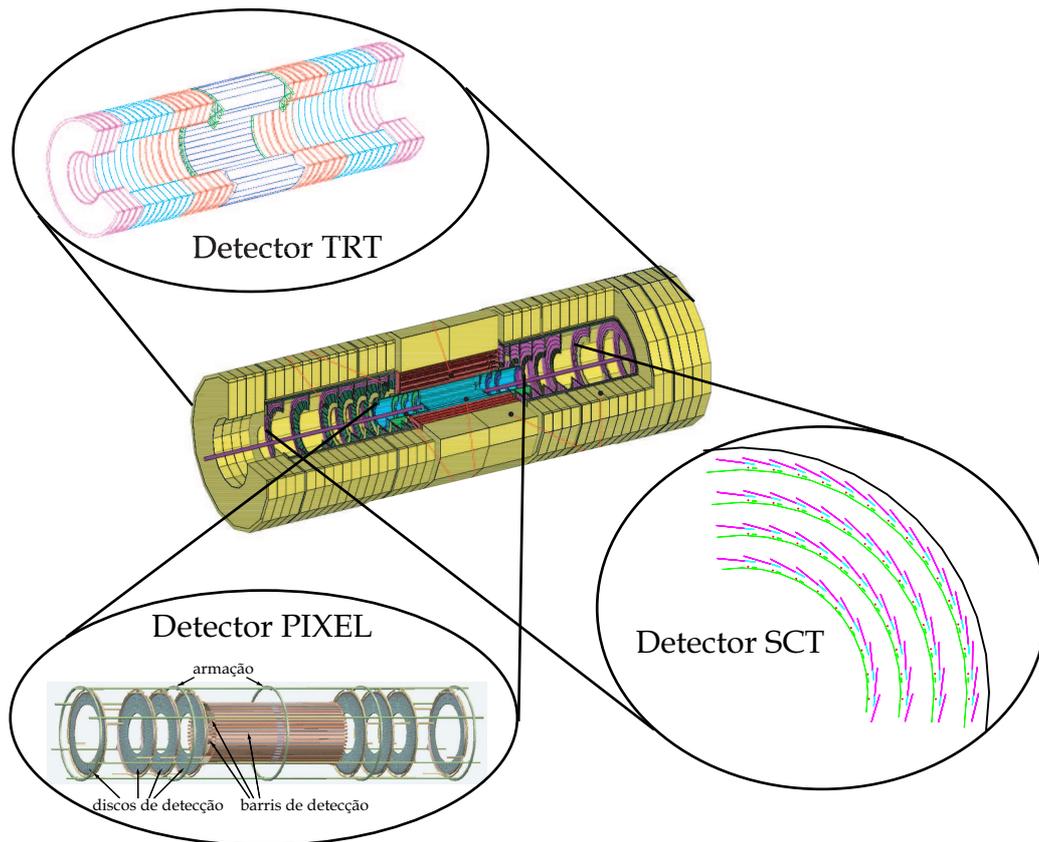


Figura 2.7: Representação tridimensional dos Detectores Internos do ATLAS.

O momento e o vértice primário de colisão requerem medidas de grande precisão, a serem feitas por detectores com fina granularidade, devido à alta densidade

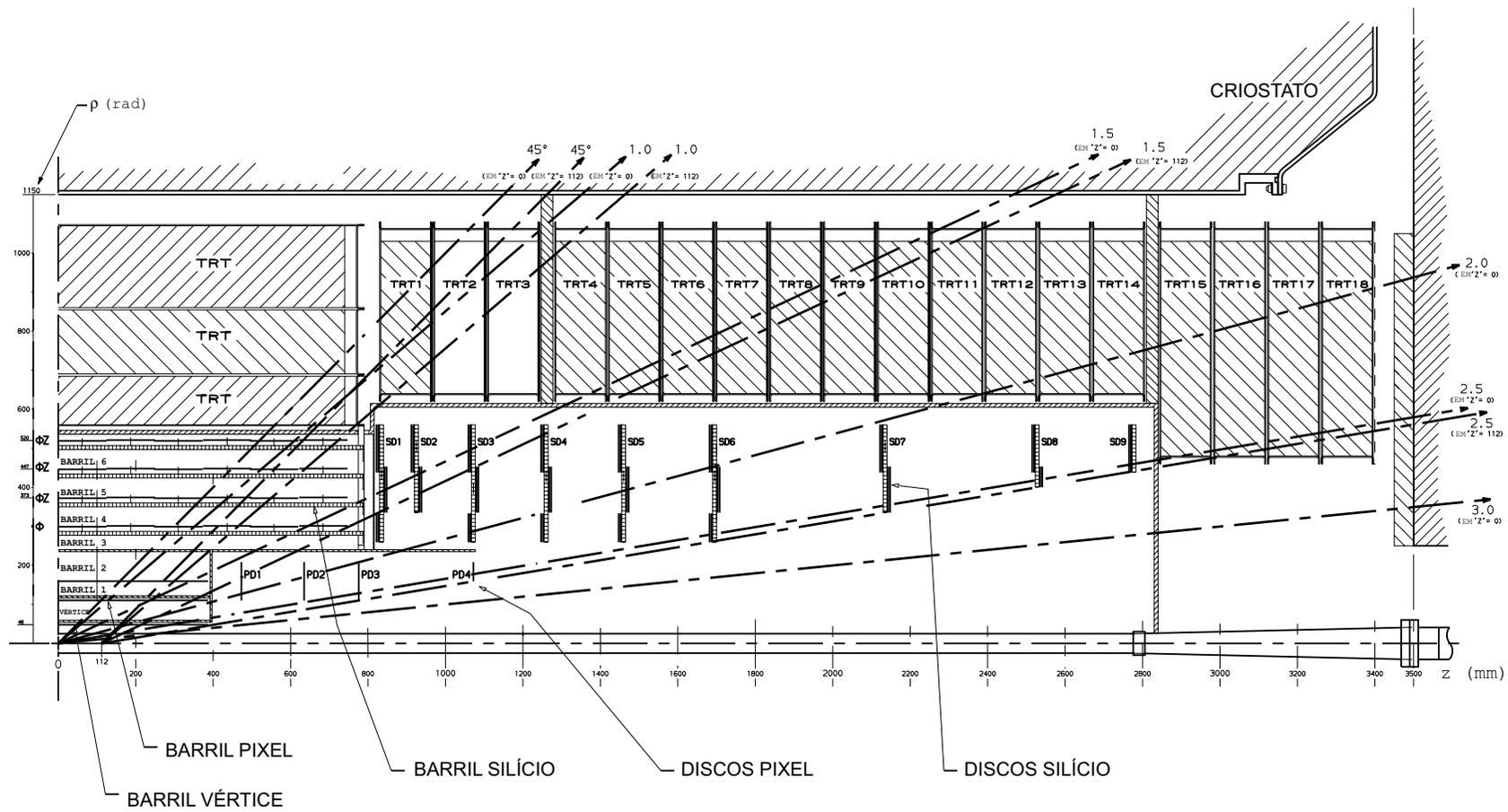


Figura 2.8: Secção transversal ( $\rho \times z$ ) da geometria dos Detectores Internos.

de trajetórias esperadas no LHC. Para realizar estas medidas, são utilizados detectores de tiras de silício (SCT), que utilizam microtubos de silício, e tecnologia de Pixel[2]. Na região mais próxima ao vértice de colisão, utilizam-se granularidades mais finas com os detectores de Pixel (PIXEL). No entanto, o número total de camadas de precisão deve ser limitado, para que não haja comprometimento das medidas dos detectores mais externos, como os calorímetros, e por causa dos elevados custos. Pelo menos quatro camadas de tiras de silício e três camadas de pixel são utilizadas na realização dos detectores internos. Um grande número de pontos de rastreamento (tipicamente 36 por trajetória) é produzido pelo rastreador em tubos (TRT), que proporcionam a possibilidade de um acompanhamento contínuo da trajetória com muito menos material por ponto e, por conseguinte, com custo menor. A combinação destas técnicas fornece um sistema de reconhecimento de padrões com alta precisão nas coordenadas  $\eta$  e  $z$ [20].

O raio mais externo dos detectores internos possui 115 cm e um total de 7 m de comprimento. Mecanicamente, os detectores internos são compostos por três partes: barris, discos (*end-caps*), como ilustra a Figura 2.8.

Nos barris, as camadas dos detectores de alta precisão são dispostas na forma de cilindros concêntricos ao redor do eixo do feixe e na região com  $|\eta| < 1$ , enquanto que os discos de detectores nas extremidades, *end-cap*, são montados de forma perpendicular ao eixo do feixe. As camadas dos detectores PIXEL e SCT são segmentadas em  $R$ ,  $\phi$  e  $z$ . Além disso, as tiras que constituem o detector SCT possuem uma inclinação de 40 mrad com relação à tangente da circunferência formada pela secção do cilindro. Os barris do detector TRT são formados por tubos com eixos paralelos entre si, distribuídos perpendicularmente à secção transversal do detector interno. Todos os elementos de rastreamento dos discos (*end-caps*), de todos os detectores internos, estão localizados de forma perpendicular ao eixo do feixe[21].

Os parâmetros básicos de projeto e a resolução para medidas de pontos no espaço (*space-points*) estão resumidos na Tabela 2.2. São apresentados valores típicos de resolução, visto que os valores reais de resolução dependem de  $|\eta|$ . A geometria fornece uma cobertura de  $|\eta| \leq 2,5$ . O tempo de vida destes detectores será limitado pelos danos de radiação, devendo ser substituídos após alguns anos[2].

Detector	Posição	Área (m <sup>2</sup> )	Resolução $\sigma(\mu\text{m})$	Canais (10 <sup>6</sup> )	Cobertura $\eta$
Pixels	1 barril	0,2	R $\phi$ = 12, z= 66	16	$\pm 2,5$
	2 barris	1,4	R $\phi$ = 12, z= 66	81	$\pm 1,7$
	4 discos <i>end-cap</i> de cada lado	0,7	R $\phi$ = 12, z= 77	43	1,7-2,5
SCT	4 barris	34,4	R $\phi$ = 16, z= 580	3,2	$\pm 1,4$
	9 discos <i>end-cap</i> de cada lado	26,7	R $\phi$ = 16, z= 580	3,0	1,4-2,5
TRT	Barril axial composto por tubos		170 (por tubo)	0,1	$\pm 0,7$
	<i>End-caps</i> de tubos radiais		170 (por tubo)	0,32	0,7-2,5

Tabela 2.2: Parâmetros do Detector Interno - Valores típicos de posição, área, resolução e cobertura de atuação dos detectores.

### 2.3.2.2 O Detector PIXEL

O detector PIXEL está sendo projetado para fornecer medidas de alta precisão, grande granularidade e o mais perto da colisão possível. O sistema fornece três medidas de precisão, e determina a resolução do parâmetro de impacto e a habilidade do detector interno em encontrar partículas de curta duração, como os b-quarks e os  $\tau$ -léptons. Os sensores de segmento bidimensionais fornecem pontos no espaço (SP, do inglês *space-points*) sem nenhuma ambigüidade associada com geometrias projetivas. No entanto, requerem a utilização de técnicas modernas de eletrônica para interconexões e leituras.

O sistema possui 140 milhões de elementos de detecção, cada um com  $50 \mu\text{m}$  na direção  $R\phi$  e  $300 \mu\text{m}$  na direção z. O sistema é composto por três barris com raio médio de  $\approx 4 \text{ cm}$ ,  $11 \text{ cm}$  e  $14 \text{ cm}$ , e quatro discos de cada lado, *end-caps*, com raios entre  $11 \text{ cm}$  e  $20 \text{ cm}$ , o que completa a cobertura angular. O sistema é projetado para ser modular, contendo aproximadamente 1500 módulos idênticos de barris e 1000 módulos idênticos de discos.

Os módulos de barris e discos do PIXEL são bastante similares. Cada módulo de barril possui  $62,4 \text{ mm}$  de comprimento e  $22,4 \text{ mm}$  de largura, com 61440 elementos pixel. A leitura é feita por 16 circuitos integrados, cada um cobrindo uma matriz de 24 por 160 pixels. Os módulos são sobrepostos na estrutura de suporte de forma a produzir uma cobertura hermética[22].

### 2.3.2.3 O Detector SCT

O sistema SCT está sendo projetado para fornecer quatro medidas de precisão por trajetória, em um alcance radial intermediário, contribuindo para medidas de momento, parâmetro de impacto e posição do vértice primário de colisão. Este detector também fornece um bom reconhecimento de padrões, através da utilização de alta granularidade. O sistema possui uma ordem de magnitude maior em área de superfície do que a geração anterior de detectores de microtubos[23]. Adicionalmente, deverá suportar níveis de radiação que irão alterar as características fundamentais do silício[24].

Os barris do SCT usam quatro camadas de detectores de microtubos de silício para fornecer pontos de precisão nas coordenadas  $R, \phi$  e  $z$ , utilizando um pequeno ângulo estéreo para obter a medida em  $z$ . Cada detector de silício possui  $6,36 \times 6,40 \text{ cm}^2$ , com 768 tiras de leitura com passo de  $80 \mu\text{m}$ . Cada módulo consiste de quatro detectores. Em cada lado do módulo, dois detectores são conectados por um fio para formar longas tiras de  $12,8 \text{ cm}$ . Dois destes pares de detectores são então dispostos com um ângulo de  $40 \text{ mrad}$ , separados por uma superfície de isolamento térmico. A eletrônica é montada em cima dos detectores.

O SCT contém  $61 \text{ m}^2$  de detectores de silício, com 6,2 milhões de canais de leituras. A resolução espacial é de  $16 \mu\text{m}$  em  $R\phi$  e  $580 \mu\text{m}$  em  $z$ . As trajetórias podem ser distinguidas quando separadas por uma distância maior do que  $200 \mu\text{m}$ , aproximadamente.

Os módulos de barris são montados em suportes locais com raios de 300, 373, 447 e 520 mm. Os módulos dianteiros são compostos por três anéis em nove discos. O limite de cobertura é de  $|\eta| \leq 2,5$ .

### 2.3.2.4 O Detector TRT

O TRT está baseado na utilização de detectores em tubos, que podem operar a taxas muito elevadas graças a seus pequenos diâmetros e ao isolamento dos elementos sensíveis dentro de dispositivos individuais de gás. Esta técnica é intrinsecamente resistente à radiação e permite um grande número de medidas a um baixo custo, tipicamente 36 medidas a serem feitas para cada trajetória. No entanto, o detector deve lidar com a alta taxa de colisões e elevada luminosidade

previstas no funcionamento do LHC.

Cada tubo possui 4 mm de diâmetro e 150 cm de comprimento. O barril contém cerca de 50000 tubos. Os *end-caps* possuem 320000 tubos dispostos de forma radial. Este detector permite a discriminação entre diversas trajetórias. A seção do barril é constituída de módulos individuais com 329 e 793 tubos cada, cobrindo uma área de 56 a 107 cm. Os dois *end-caps* consistem de 18 discos próximos do ponto de colisão, cobrindo uma faixa de 64 a 103 cm.

O TRT contribui para a acurácia na medida de momento do Detector Interno, através de um conjunto de medidas de 50  $\mu\text{m}$  de precisão. Ele adiciona ao sistema de reconhecimento de padrões cerca de 36 pontos por trajetória e permite que métodos simples e rápidos de reconstrução de trajetórias sejam implementados.

### 2.3.3 O Sistema de Filtragem do ATLAS

O sistema de filtragem e aquisição de dados do ATLAS (sistema *Trigger-DAQ*) é baseado em três níveis de seleção de eventos em tempo real[25, 26, 27]. Cada nível de filtragem refina a decisão realizada no estágio anterior e, quando necessário, aplica critérios adicionais de seleção. Partindo de uma taxa de colisão de 40 MHz, a taxa de seleção de eventos deve ser reduzida a cerca de 100 Hz para armazenamento. Este processo exige um fator de rejeição de  $\approx 10^7$  eventos e deve garantir excelentes taxas de eficiência na detecção de processos físicos raros como, por exemplo, o decaimento do bóson de Higgs. A Figura 2.9 apresenta uma visão funcional simplificada do sistema de *Trigger-DAQ*[28]. Na seqüência, uma breve descrição de alguns aspectos chaves para o processo de seleção é apresentada.

O primeiro nível de filtragem (LVL1)[25] realiza a seleção inicial, baseado em informações dos calorímetros e dos detectores rápidos de Múons. A seleção dos calorímetros<sup>2</sup> é baseada em informações de granularidade reduzida. Os objetos procurados pelos sistemas de filtragem possuem elevado momento transverso.

Durante o tempo de processamento do LVL1, as informações de todos os canais dos detectores são armazenadas em memórias, em *pipeline*. O número total

---

<sup>2</sup>Os calorímetros são equipamentos responsáveis por medir a energia de partículas, sendo capazes de produzir uma resposta rápida e de permitir a discriminação eficiente das partículas que com ele interagem.

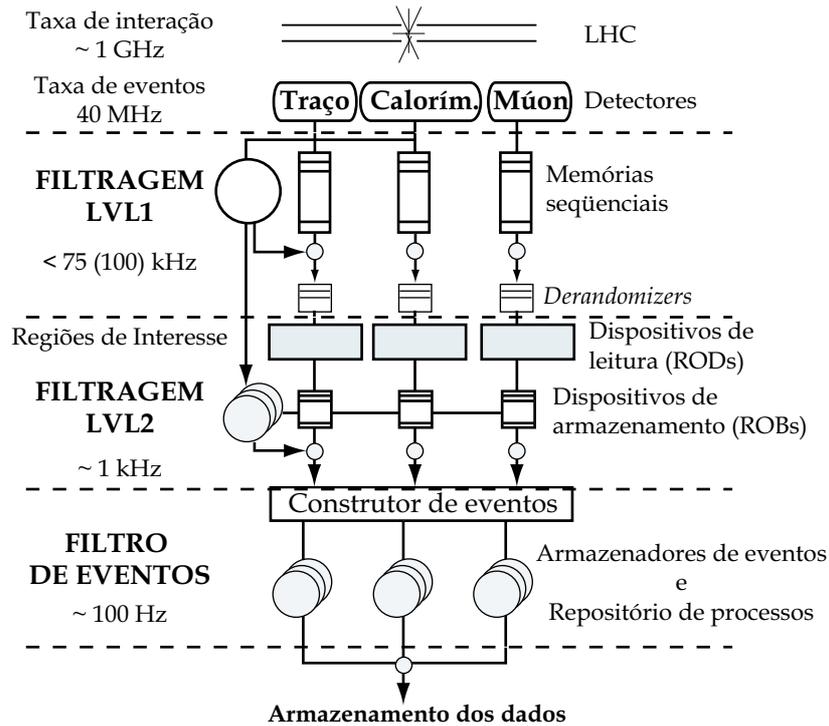


Figura 2.9: Esquema do sistema de filtragem do ATLAS (TDAQ).

de canais é de  $10^7$ . A latência do LVL1 medida desde a colisão próton-próton até a disponibilização da decisão de filtragem é de  $2,5 \mu\text{s}$ . De forma a atingir esta meta, o sistema está sendo baseado em lógica de baixa programabilidade (ASICs e FPGAs)[29]. Os eventos selecionados pelo LVL1 são lidos pelos dispositivos de leitura (RODs) e armazenados nas memórias de leitura (ROBs).

Todos os dados armazenados nas ROBs são analisados pelo segundo estágio de filtragem (LVL2). Caso o evento armazenado seja considerado relevante pelo LVL2, ele é transferido para o estágio de Filtragem de Eventos (EF)[26]. O processo de mover os dados dos ROBs para o EF é chamado de construção de eventos.

Os estágios LVL2 e EF fazem parte do Sistema de Filtragem de Alto Nível ou *High Level Trigger* (HLT). A Seção 2.4 apresenta os critérios de seleção, os objetos de interesse, a arquitetura e a implementação deste sistema de seleção.

## 2.4 O Sistema HLT (*High Level Trigger*)

O sistema HLT é representado pelos segundo e terceiro estágios de seleção de eventos. Ele é composto por três partes principais: o sistema LVL2, o filtro de

eventos (EF) e o *software* de seleção de eventos (ESS)[30].

O filtro LVL2 está sendo construído de forma a utilizar o mecanismo de seleção de regiões de interesse (RoI) apresentado pelo filtro LVL1. Isto inclui a informação da posição  $\eta$  e  $\phi$  e o momento transversal  $p_t$  do conjunto de dados. Utilizando a informação de localização da RoI, o LVL2 acessa os dados dos ROBs de forma seletiva, movendo somente os dados relevantes para a decisão deste nível. Caso seja necessário, o LVL2 tem acesso ao conjunto de dados completo com total precisão e granularidade.

É esperado que o LVL2 reduza a taxa de processamento para aproximadamente 1 kHz. A latência do filtro LVL2 varia conforme o evento e espera-se estar na faixa de 1 a 10 ms. De modo a realizar esta tarefa, o LVL2 conta com processadores comercialmente disponíveis (como computadores pessoais - PCs) conectados através de uma tecnologia de rede adaptada a esse propósito. Cada RoI será examinada no detector na qual foi originada (como, por exemplo, nos calorímetros) de forma a confirmar a validade do objeto. Após a confirmação, outras características são procuradas em outros detectores, como os de traço, por exemplo.

Depois do estágio LVL2, o último estágio de seleção *online* é o Filtro de Eventos (EF). Este estágio emprega métodos e algoritmos desenvolvidos para o processamento em *offline*, adaptados ao ambiente de execução *online*. O EF irá fazer a seleção final dos eventos de física que serão armazenados para análise subsequente *offline*. Este estágio deve reduzir por uma ordem de magnitude a saída do LVL2, ou seja,  $\approx 100$  Hz. O poder de rejeição do EF consiste em:

- utilizar algoritmos refinados e, quando necessário, apertar o nível de decisão de  $p_t$  comparado com aqueles utilizados no LVL2;
- dispor de todos os dados relevantes de um determinado evento nos cálculos e critérios de seleção;
- utilizar algoritmos com critérios complexos que não puderam ser utilizados no LVL2 devido ao tempo de processamento. Um exemplo é a determinação do vértice primário de colisão e o rastreamento de trajetórias utilizando a recuperação de *bremstrahlung* de elétrons.

### 2.4.1 Os Critérios de Seleção

De forma a garantir boa aceitação dentro do paradigma atual da física de partículas, uma abordagem baseada em enfatizar o critério de seleção *online* foi adotada. Isto significa filtrar assinaturas baseadas em objetos com elevado momento ( $p_T$ ). Quanto mais exclusiva é a seleção dos elevados  $p_T$ , mais criteriosa é a topologia dos diversos “objetos físicos“ (múons, elétrons, jatos, etc.).

A seleção do HLT, na maior parte dos casos, é realizada pela informação obtida no LVL1 e irá explorar características adicionais da filtragem do LVL2 e do EF. No LVL2, uma rápida rejeição deve ser alcançada, utilizando algoritmos dedicados e que trabalham dentro das restrições de latência. Esses algoritmos irão, na maior parte dos casos, utilizar uma baixa porcentagem dos dados do evento, graças ao critério de seleção de regiões de interesse apontadas pelo LVL1. A seleção de assinaturas será refinada no EF, onde o evento completo estará disponível para análise. O estágio EF utiliza parâmetros de alinhamento e calibração mais detalhados, possuindo maior precisão e latência de execução.

### 2.4.2 Objetos de Física para Seleção de Eventos

O sistema de filtragem do ATLAS baseia-se no conceito de “objetos físicos“. Tipicamente os objetos candidatos são identificados e reconstruídos de forma simples pelo LVL1. O processamento no HLT irá progressivamente refinar a reconstrução, rejeitando objetos falsos e melhorando a precisão nos parâmetros de medida.

Os objetos de física usados no HLT podem estar baseados na informação de todos os subdetectores do ATLAS, com granularidade completa. Conforme mencionado anteriormente, a diferença na reconstrução dos objetos entre o LVL2 e o EF refere-se, predominantemente, à complexidade dos algoritmos em interpretar os dados brutos, e de acordo com o nível de detalhe e acurácia da informação de calibração e alinhamento utilizada. O EF possui o evento completo a sua disposição para buscar por esses objetos, mesmo que neste estágio sejam utilizadas em sua inicialização as respostas dos algoritmos do LVL2.

Os objetos de física mais importantes que o sistema HLT deve reconstruir, são: elétrons; fótons; múons; taus; jatos; jatos b; energia transversa  $E_T$ ; energia total  $\sum E_T$ .

### 2.4.3 Arquitetura do Sistema HLT

A arquitetura do HLT está baseada em módulos, com funções distintas, que são implementados de forma independente. O sistema HLT proporciona ao experimento ATLAS a capacidade de selecionar os eventos que são de interesse para o estudo da física, controlar e monitorar todo o experimento.

As seguintes funções são identificadas:

- **Leitura dos detectores:** os dados do LVL1 são organizados por subdetector de forma fragmentada. Desta forma, o HLT pode ter acesso rápido a um determinado módulo do detector sem ter que percorrer todos eles para obter os dados do evento;
- **Movimento dos dados do evento:** os fragmentos de dados lidos são movidos para o HLT. Após o processamento estes serão armazenados em um banco de dados;
- **Seleção de eventos e armazenamento:** o sistema HLT é responsável pela redução da taxa de eventos e seleção daqueles de potencial interesse para análise *offline*. Os eventos selecionados pelo HLT serão escritos em dispositivos de armazenamento permanente para uma posterior reconstrução e análise;
- **Controle:** os controles em tempo real do TDAQ incluem a capacidade de configurar, operar e controlar o experimento durante a tomada de dados;
- **Monitoramento:** o monitoramento em tempo real inclui a capacidade de verificar o estado e a eficiência de todos os detectores do ATLAS durante a calibração, teste e execução do experimento.

### 2.4.4 Implementação da Arquitetura

Mesmo com algoritmos diferentes o LVL2 e o EF utilizam a mesma arquitetura de programa para a seleção de eventos. Isto facilita a utilização de uma infraestrutura comum e simplifica o desenvolvimento dos algoritmos no HLT. Esta arquitetura comum é chamada de *Software* de Seleção de Eventos (ESS).

O ESS consiste na infra-estrutura dos algoritmos de seleção. Estes algoritmos são fornecidos pelo grupo PESA (*Physics and Event Selection Architecture*) ou, em alguns casos, pelo grupo de reconstrução do EF.

No sistema HLT, o ESS será executado pelo ambiente de *software* proporcionado pelo L2PU (*LVL2 Processing Unit*) e pelo PT (*Processing Task*) do EF. Desta forma, o ESS necessita estar de acordo com as restrições do ambiente em tempo real, o que significa, estar dentro dos objetivos de eficiência e latência em tempo real.

É desejado também que o ESS seja executado diretamente do ambiente de desenvolvimento *offline*, ATHENA. Isto facilita o desenvolvimento dos algoritmos, o estudo das fronteiras entre EF e LVL2 e permite estudos de eficiência para a análise física. Desta forma, o ESS precisa ter o controle da infra-estrutura e dos serviços que são fornecidos pela arquitetura de *software offline*. Por esta razão o ambiente de desenvolvimento ATHENA foi escolhido como o ambiente de execução do ESS no PT do EF e no L2PU do LVL2.

A tarefa do ESS consiste em emular um processo completo de seleção em tempo real. Para tal, três algoritmos de alto nível do ATHENA são necessários, que são: a emulação da filtragem do LVL1 e a duas instâncias do *software* de seleção de eventos. A emulação da filtragem do LVL1 fornece o resultado do LVL1. A primeira instância do *software* de seleção de eventos executa o LVL2 alimentado pelo resultado do LVL1, e a segunda instância executa a seleção do EF alimentada pelo resultado do LVL2.

#### **2.4.4.1 Ambiente de Desenvolvimento ATHENA**

De modo a paralelizar o desenvolvimento dos diversos estágios de construção do ATLAS, foi desenvolvido o ATHENA[31], um sistema de simulação, reconstrução e visualização. O objetivo consiste em disponibilizar ferramentas que facilitem a análise e escrita de programas para os sistemas *online* e *offline* do ATLAS.

Esse ambiente de desenvolvimento possui ferramentas para realizar simulações de uma colisão próton-próton, simulação da resposta dos detectores, simulação da resposta do LVL1, LVL2 e EF. A arquitetura consiste na especificação de um número de componentes e suas interações. Cada componente é um bloco de programas que

possui interfaces e funcionalidades padronizadas.

Os componentes principais da arquitetura de programas ATHENA está representado no diagrama da Figura 2.10. Este diagrama de objetos explica como o sistema é composto. Ele representa um estado hipotético, mostrando objetos e suas relações, não representando a estrutura hierárquica do programa.

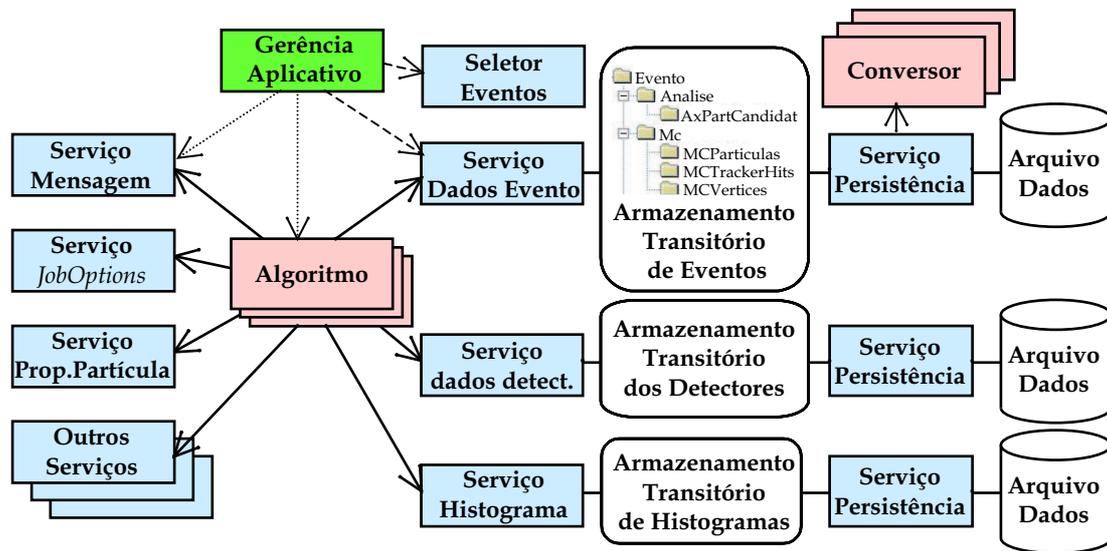


Figura 2.10: Diagrama da arquitetura de objetos do ATHENA.

Os programas escritos para geração de eventos, reconstrução ou análise, consistem em especializações de alguns componentes específicos. Desta forma, a elaboração de códigos se dá sob a forma de algoritmo, objeto de dados ou conversores. Um código pode ser utilizado somente em determinados eventos ou várias vezes em um mesmo evento. Este último tipo é chamado de ferramenta ou serviço.

É certamente importante decompor um grande sistema de programas em hierarquias com entidades menores e mais fáceis de manipular. Esta decomposição pode ter conseqüências importantes para implementação como, por exemplo, dependências de outros códigos, tempo de compilação, etc.

A arquitetura de desenvolvimento ATHENA está sendo construída em C++. Todo o ambiente de desenvolvimento do sistema de filtragem (*trigger*) está sendo desenvolvido nesta plataforma.

## Capítulo 3

# Reconstrução de Vértices - Estado da Arte

Atualmente existem em torno de vinte e seis grupos que investigam detectores de pixel e de silício, com o objetivo de encontrar o vértice primário de colisão de aceleradores lineares e circulares[32]. Este capítulo apresenta, de forma resumida, as técnicas de reconstrução de vértices que foram estudadas ao longo dos anos e a nível mundial dando ênfase àquelas que estão sendo utilizadas atualmente nos quatro detectores do CERN que operarão no LHC.

Em um pouco mais de duas décadas de estudos, aproximadamente trinta variações de localização tridimensional do vértice primário de colisão foram testadas em pelo menos onze tecnologias diferentes de detectores de Pixel e de Silício. Técnicas de mínimos quadrados (*least square* - LS)[33, 34, 35], e filtragem de Kalman[36] foram utilizadas para estimar o vértice de colisão de forma *offline*. Estudos de otimização dos filtros de Kalman foram realizadas de forma a utilizar os métodos de localização do vértice de colisão de forma *online*[37, 38, 39, 40]. Além destas técnicas outras como, por exemplo, filtros de somas gaussianas[4], recursões determinísticas (*deterministic annealing*)[5] e mínimos médios quadráticos (*least mean square* - LMS)[41] podem ser destacadas.

Diversos estudos de localização de vértices, tanto primários quanto secundários, utilizavam a resposta dos algoritmos de reconstrução de trajetórias para determinar com precisão a localização dos vértices de colisão[37, 3]. Devido a este fato, é possível notar a semelhança entre as técnicas de localização de vértices e de

reconstrução de trajetórias[42, 33, 43]. A otimização das técnicas de localização de vértices possibilitou sua utilização *online* e antes dos algoritmos de reconstrução de trajetórias. Desta forma, os métodos de localização de vértices funcionam como um filtro, reduzindo significativamente o tempo de execução de dos métodos reconstrução das trajetórias[44, 45, 39, 46].

As próximas seções apresentam um resumo dos estudos de localização de vértices e reconstrução de trajetórias realizados nos experimentos de colisão de partículas.

### 3.1 Sistemas de Localização de Vértices Primários

Esta seção apresenta um resumo das técnicas encontradas neste estudo sobre o estado da arte da localização de vértices primários. Ao todo oito dos onze métodos de localização de vértices primários de colisão estão resumidas nas subseções a seguir.

Devido ao fato dos métodos terem sido utilizados em detectores diferentes, uma pequena contextualização ao ambiente de detecção é apresentada. O algoritmo *ZFinder*, utilizado como comparação de performance para os métodos de transformada de Hough, encontra-se na seção 3.2.1.

A transformada de Hough fora utilizada em sistemas de reconstrução de trajetórias[47] do detector SFT (*Scintillating Fibre Tracker*) no acelerador LEP<sup>1</sup> no CERN. A idéia de utilizar a transformada de Hough foi motivada pela aplicação deste estudo no algoritmo *ZFinder*. Isto se deve ao fato do algoritmo *ZFinder* possuir uma abordagem próxima a de Hough para ajuste de retas, como será observado na seção 3.2.1.

#### 3.1.1 Localização 3D por Correlação Cruzada

A reconstrução tridimensional do vértice primário de colisão é realizada pela correlação cruzada dos pontos do detector PIXEL do detector ALICE[1]. No experimento ALICE, a região do vértice de colisão é parametrizada por uma gaussiana de média zero ao longo dos eixos  $x$ ,  $y$  e  $z$ . Nos eixos  $x$  e  $y$ , a gaussiana possui desvio

---

<sup>1</sup>*Large Electron-Positron Collider*. Colisionador elétron-pósitron do CERN. Atualmente desativado para a construção do LHC.

padrão de  $\sigma_{x,y} \approx 75\mu m$ . No eixo  $z$ , no entanto, a gaussiana possui um desvio padrão de  $\sigma_z = 5,3cm$ .

O conhecimento da localização do vértice ao longo do eixo do feixe pode reduzir significativamente o ruído de fundo dos eventos. A técnica consiste na correlação cruzada dos pontos nas duas camadas do detector PIXEL, em uma região de  $4\sigma_z$  em torno da localização estimada do vértice  $z_v$  e um corte em  $\Delta\phi$  dependente do campo magnético externo aplicado.

Foi utilizado um algoritmo recursivo, começando por uma primeira aproximação do vértice, calculando a posição correta pela correlação entre os pontos das duas camadas do detector PIXEL e, em seguida, repetindo este procedimento. Verificou-se a utilização de 2 ou 3 iterações do algoritmo, uma vez que a condição de parada dada era a aproximação dos pontos a uma linha reta. A Figura 3.1 apresenta as distribuições de  $x_v$  e  $y_v$  dos vértices obtidas. A largura da distribuição é originada pela aproximação a uma reta.

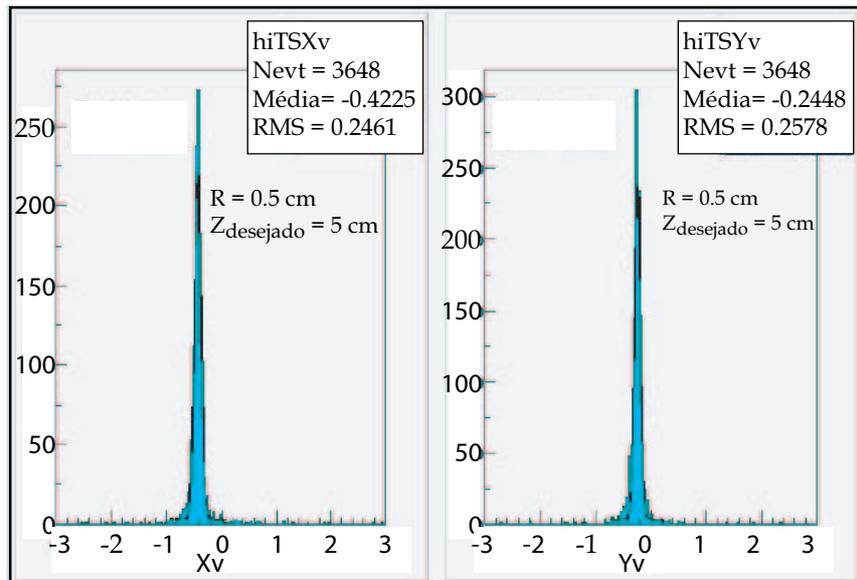


Figura 3.1: Distribuição do vértice nos eixos  $x$  e  $y$ . Extraído de [1].

O cálculo do centróide das distribuições de  $x_v$ ,  $y_v$  e  $z_v$  possibilita a localização do vértice primário nestas três dimensões. A Figura 3.2 apresenta os resultados ao longo dos eixos  $x$ ,  $y$  e  $z$ , as diferenças entre a posição do vértice simulado (*true*) e o obtido são apresentadas. A obtenção destes resultados foi realizada para vértices gerados de forma aleatória no plano  $xy$  e com distâncias radiais entre 0 e 10 mm.

A resolução desta técnica, como observa-se pela Figura 3.2 está na ordem de

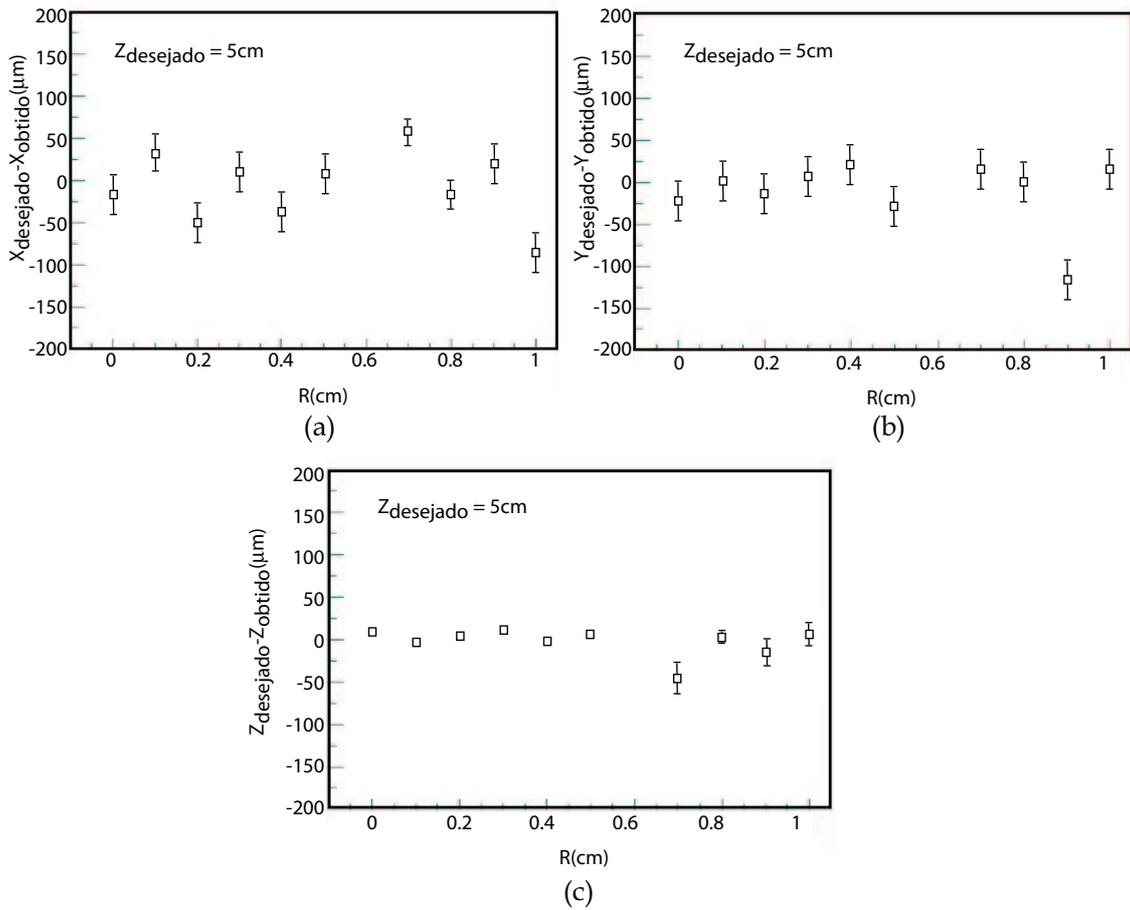


Figura 3.2: Diferença entre o valor desejado e o obtido em função da distância radial da localização do vértice com relação ao eixo  $z$ . (a)  $x_{true}$  e  $x_{found}$ ; (b)  $y_{true}$  e  $y_{found}$ ; c)  $z_{true}$  e  $z_{found}$ . Extraído de [1].

poucos  $\mu m$  ao longo do eixo  $z$ , enquanto é um pouco mais larga ao longo dos eixos  $x$  e  $y$ .

### 3.1.2 Localização por Filtragem de Kalman

O algoritmo *VKalVrt*[40] apresenta uma aplicação do método de filtragem de Kalman para localização do vértice primário de colisão do ATLAS. O experimento ATLAS utiliza três abordagens diferentes para a reconstrução do vértice primário de colisão. Duas utilizam parâmetros de hélices e a terceira utiliza aproximação por retas. Os grupos envolvidos são o de Reconstrução, o de Física Aplicada (ATC) e o de Física e Arquitetura de Seleção de Eventos (PESA).

O grupo de Reconstrução utiliza as equações de hélices para localizar os vértices pelo método de filtragem de Kalman[40]. O grupo de Física Aplicada também utiliza as equações de hélice e a reconstrução é feita pelo método de Billoir[48]. O grupo PESA utiliza um ajuste linear dos pontos, aproximando hélices por retas[49].

A reconstrução tridimensional realizada pelo algoritmo *VKalVrt* localiza a posição do vértice e calcula o momento da trajetória. As informações de entrada para o ajuste consiste em cinco parâmetros da trajetória de partículas. Para uma trajetória 3D curva, submetida a um campo eletromagnético externo, além destes cinco parâmetros, é utilizada uma matriz de pesos  $W_i$  no ponto de referência de cada trajetória.

A parametrização por hélice possibilita a extrapolação da trajetória da partícula para obtenção da região de colisão. Assume-se que uma partícula de vértice  $\vec{V}$ , com parâmetros da trajetória  $q_i$  pode ser expressa como uma função de  $\vec{V}$  e do momento  $\vec{p}_i$ :  $\mathbf{q}' = \mathbf{T}(\vec{V}, \vec{p}_i)$ .

$$\chi^2 = \sum_{N_{track}}^{i=1} (\mathbf{q}_i - \mathbf{T}(\vec{V}, \vec{p}_i))^T W_i (\mathbf{q}_i - \mathbf{T}(\vec{V}, \vec{p}_i)) \quad (3.1)$$

O ajuste funcional do vértice é realizado pela equação 3.1, onde  $W_i$  é uma matrix de pesos de dimensão  $5 \times 5$  do ajuste da trajetória. O objetivo do ajuste consiste em encontrar  $\vec{V}$  e  $\vec{p}_i$  que minimizem a equação 3.1. O momento da partícula no vértice ( $\vec{p}_i$ ) é ajustado pela comparação com o momento obtido pela reconstrução ( $\mathbf{q}_i$ ).

O valor de  $\chi^2$  da equação 3.1 pode ser linearizado em pontos próximos ao vértice, resultando na equação 3.2.

$$\chi^2 = \sum_{N_{track}}^{i=1} (\delta \mathbf{q}_i - D_i \delta \vec{V} - E_i \delta \vec{p}_i)^T W_i (\delta \mathbf{q}_i - D_i \delta \vec{V} - E_i \delta \vec{p}_i) \quad (3.2)$$

onde  $D_i = \frac{\partial \mathbf{T}(\vec{V}, \vec{p}_i)}{\partial \vec{V}}$  e  $E_i = \frac{\partial \mathbf{T}(\vec{V}, \vec{p}_i)}{\partial \vec{p}_i}$  são matrizes das derivadas. A representação mais precisa destas matrizes derivadas pode ser encontrada em [40].

Nesta abordagem, o filtro de Kalman consiste em recalcular a posição do vértice a cada trajetória adicionada. Neste caso a equação de Kalman é representada por 3.3 e 3.4.

$$\vec{V}_i = \vec{V}_{i-1} \quad (3.3)$$

$$\begin{aligned} \mathbf{q}_i &= \mathbf{T}(\vec{V}, \vec{p}_i) + \varepsilon_i \\ &= \mathbf{c}_{0i} + D_i \vec{V}_i + E_i \vec{p}_i + \varepsilon_i \end{aligned} \quad (3.4)$$

onde  $\varepsilon_i$  é o ruído do processo. O método de filtragem de Kalman (equações 3.3 e 3.4) e o método de ajuste através da equação 3.2 realizam de forma precisa o cálculo do vértice de colisão, a atualização dos parâmetros de trajetória e as matrizes de erro.

Este método pode ser utilizado para reconstrução de vértices primários e secundários, em aplicações de diferentes restrições. O vértice de reconstrução primário demora  $\approx 1$  ms para eventos de elétrons individuais e  $\approx 3$  ms para eventos de elétrons com empilhamento, em um Pentium III (1GHz). Este método é utilizado em aplicações de *b-tagging*[48] com estudo de vértices secundários.

### 3.1.3 Localização pelo Método de Billoir

O método de Billoir[3] consiste em uma variação da filtragem de Kalman. Este método foi inicialmente utilizado no experimento ZEUS em 1990[50, 43]. Atualmente ele está sendo desenvolvido pelo grupo ATC. Encontra-se codificado em C++ no ambiente de desenvolvimento ATHENA.

O ponto de partida do algoritmo está na utilização de todos os pontos que pertencem às trajetórias reconstruídas posicionados na camada mais externa do

detector de traço. Um ponto da camada mais externa com um vizinho a ele na camada imediatamente abaixo formam a primeira tentativa de trajetória. A janela de vizinhança depende do momento da partícula, ou seja, quanto menor o momento maior a janela. Na seqüência, é adicionado mais um ponto de uma camada mais interna, de forma que esta recursão ocorra até a primeira camada e a trajetória seja extrapolada para o vértice.

A Figura 3.3 apresenta a projeção  $xy$  do detector de traço. A Figura 3.4, por sua vez, apresenta o mecanismo de rastreamento de trajetórias, onde  $x_n^m$  representa as coordenadas de medida da  $n$ -ésima camada;  $x_n^F$  representa as coordenadas da trajetória ajustada;  $x_n^{F_c}$  denota a coordenada de extrapolação de  $x_n^F$  na camada  $(n+1)$ . Os valores 1, 2 e 3 são os passos realizados pelo algoritmo.

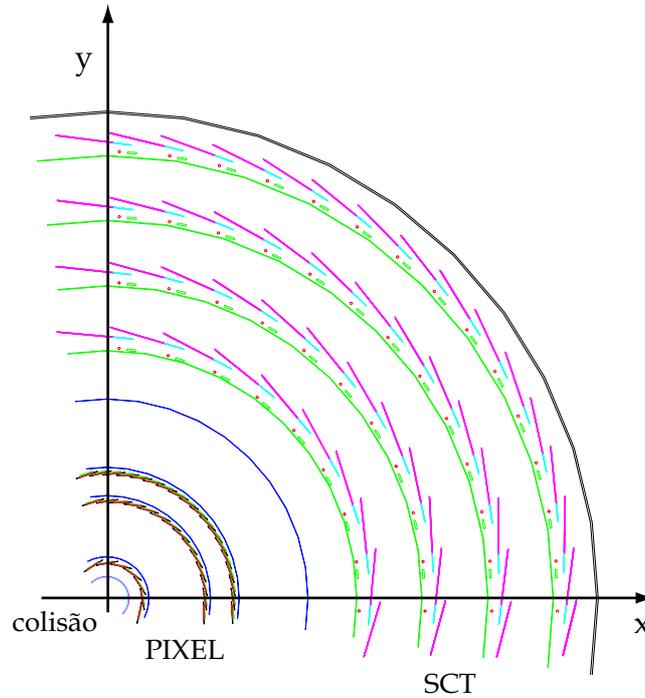


Figura 3.3: Vista  $x \times y$  dos detectores de traço do ATLAS. Extraído de [2].

O método ajusta os pontos do evento pelo método de Billoir[3] e retorna o vértice e a lista de trajetórias utilizadas. Com esta informação, o sistema de localização de vértices realiza uma segunda seleção de trajetórias baseada no  $\chi^2$  (equação 3.5) da trajetória no ajuste para eliminação de ruído. Trajetórias com contribuições de  $\chi^2 > 4$  são removidas e a localização do vértice é refinada pelas trajetórias que restaram.

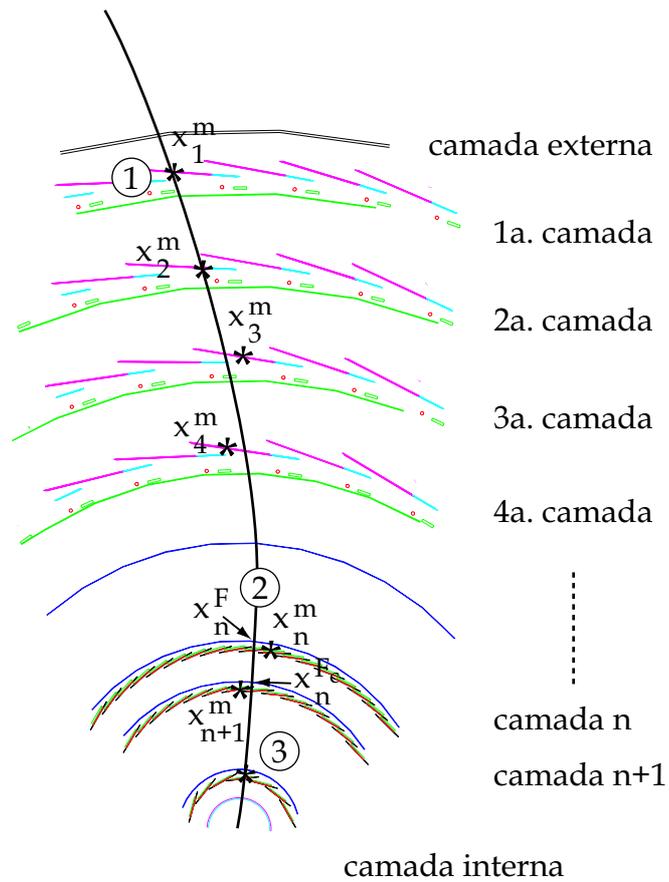


Figura 3.4: Seqüência de rastreamento de trajetórias.  $x_n^m$  representa a coordenada na  $n$ -ésima camada que pertence a trajetória reconstruída. Extraído de [3].

$$F(V^0 + \delta V, \mathbf{p}_i^0 + \delta \mathbf{p}_i) = F(V^0, \mathbf{p}_i^0) + \mathbf{D}_i \delta V + \mathbf{E}_i \delta \mathbf{p}_i$$

$$\chi^2 = \sum_i (\mathbf{q}_i^{meas} - F(V, \mathbf{p}_i))^T \mathbf{W}_i (\mathbf{q}_i^{meas} - F(V, \mathbf{p}_i)) \quad (3.5)$$

A resolução obtida por este método está na ordem de  $\mu m$ . Vértices reconstruídos de simulações de  $WH(120) \rightarrow \mu\nu u\bar{u}$  sem empilhamento, representados na Figura 3.5, possuem resolução em  $xy$  de  $14 \mu m$  e de  $37 \mu m$  na direção  $z$ . A Figura 3.6 apresenta a reconstrução de vértices primários de  $H(130) \rightarrow llll$  com empilhamento e alta luminosidade. A resolução é um pouco diferente, sendo igual a  $16 \mu m$  para  $xy$  e  $46 \mu m$  na direção  $z$ . Em 88% dos casos o vértice primário reconstruído está a uma distância tridimensional de  $0,5 \text{ mm}$  do valor simulado nestes eventos de empilhamento a alta luminosidade.

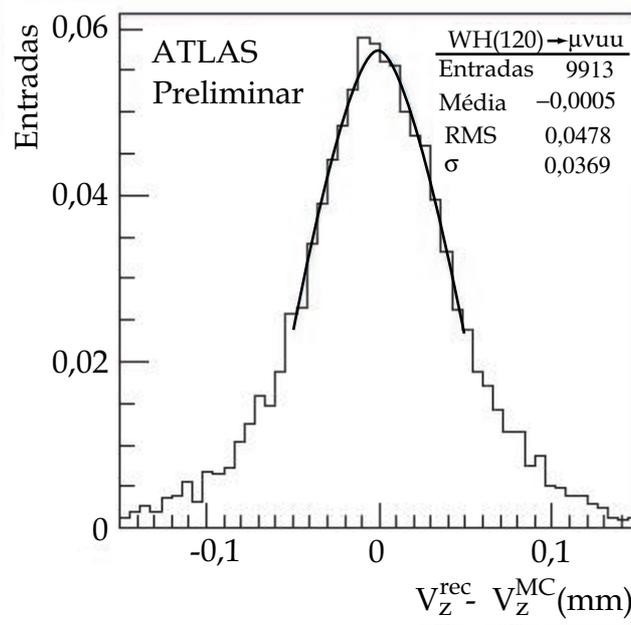


Figura 3.5: Resolução dos vértices primários reconstruídos de eventos na direção  $z$  do processo  $WH(120) \rightarrow \mu\nu u\bar{u}$  sem empilhamento. Extraído de [3].

### 3.1.4 Localização por Filtros de Somas Gaussianas

O filtro por Somas Gaussianas (*Gaussian-Sum Filter* - GSF) utiliza um conjunto de gaussianas para estimar as quantidades modeladas e as medidas de erro. Este método consiste em um estimador não-linear, onde os pesos das componentes do filtro dependem da medida.

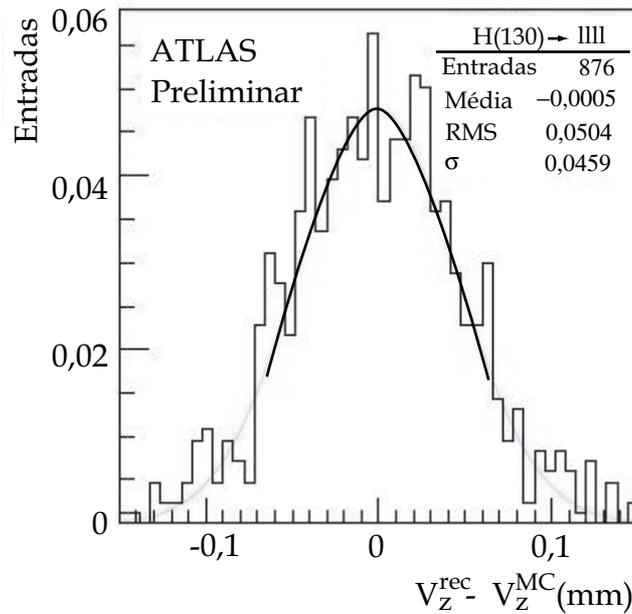


Figura 3.6: Resolução dos vértices primários reconstruídos de eventos na direção  $z$  do processo  $WH(130) \rightarrow lll$  com empilhamento e a alta luminosidade. O vértice primário correto foi encontrado em 88% dos casos. Extraído de [3].

Esta abordagem foi inicialmente utilizada para reconstrução de trajetórias[51], onde foi observado que ela possuía menor variância que o filtro de Kalman. Em particular, a GSF é bastante eficiente na reconstrução de elétrons, onde a distribuição Bethe-Heitler de radiação *bremsstrahlung* é não gaussiana e pode ser modelada por um conjunto de gaussianas.

Assim como no filtro de Kalman, a reconstrução de vértices primários é realizada de forma recursiva, onde o vértice estimado é atualizado a cada trajetória. Quando uma trajetória é adicionada ao vértice, cada componente do estado do vértice GSF é atualizada com a componente da trajetória medida pelo filtro de Kalman. Desta forma, o número de componentes da estimativa do vértice cresce exponencialmente, pois cada passo é multiplicado pelo número de componentes da nova trajetória. O GSF é a soma ponderada de diversos filtros Kalman executados em paralelo, e somente os pesos das componentes são calculados separadamente.

Para validar este algoritmo uma simulação de eventos foi criada, onde cada evento possui um vértice composto por quatro trajetórias. Não foram realizadas reconstruções de trajetórias nos eventos, e os parâmetros das trajetórias foram escolhidos de acordo com um modelo de duas componentes de um conjunto de gaussianas[4].

Os testes foram repetidos com diversas variações no modelo. De modo a comparar o método GSF com o de Kalman foram utilizadas duas janelas de intervalos simétricos, uma com 50% da distribuição residual e outra com 90% da distribuição.

A Figura 3.7 apresenta o resultado obtido nestas simulações. Conforme esperado, vértices localizados pela GSF mostraram-se mais precisos do que aqueles reconstruídos por Kalman. A eficiência relativa máxima é obtida quando a distância entre as duas componentes do conjunto de gaussianas e a da gaussiana simples é máxima, medidas pela distância de Kullback-Leibler[52]. Com o aumento na proporção de ruído, a eficiência relativa decresce, conforme o erro referente aos parâmetros da trajetória começa a dominar.

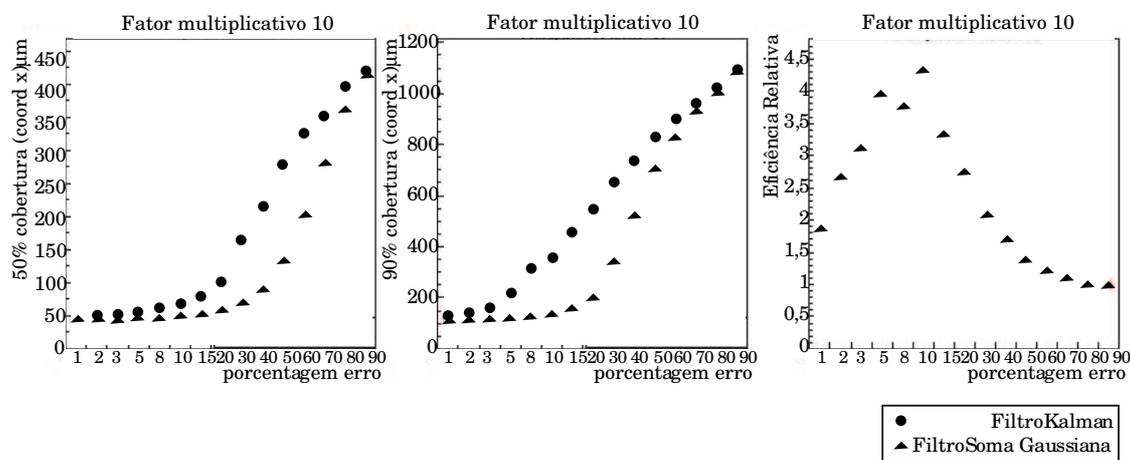


Figura 3.7: Cobertura de 50% (esquerda), 90% (meio) e eficiência relativa (direita) dos ajustes dos vértices para trajetórias com duas componentes. Extraído de [4].

### 3.1.5 Localização por Filtros de *Annealing* Determinísticos

O método de Recursão Determinística (*Deterministic Annealing*, DA) é utilizado para encontrar o melhor conjunto de pontos que formam uma trajetória (*clustering*). Este processo é bastante utilizado em problemas de reconhecimento de padrões, classificação, compressão e regressão estatística. Para a localização do vértice de colisão, este método oferece uma importante característica, a habilidade de localizar vértices em um ambiente ruidoso sem um conhecimento prévio de quantos vértices devem ser localizados.

O método DA foi implementado no CMS na plataforma ORCA[53]. O passo inicial do algoritmo consiste em caracterizar a topologia do sistema. Para tal, é

utilizado o método de aglomeramento de pontos Apex (*Apex Point*)[54], onde cada trajetória é substituída por um ponto representativo de sua densidade. A Figura 3.8 ilustra o processo DA para o aglomeramento de um conjunto de pontos.

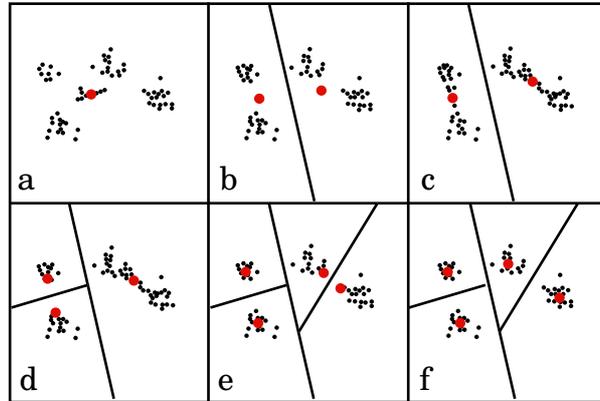


Figura 3.8: Ilustração do processo de recursão determinística (DA) para o aglomeramento de um conjunto de pontos. Extraído de [5].

O algoritmo DA consiste nos seguintes passos:

- Dado um conjunto de trajetórias reconstruídas, calcular todos os pontos Apex utilizando o algoritmo MTV[54];
- DA é aplicado ao conjunto de pontos Apex. No início um protótipo simples é utilizado para todos os pontos Apex. Com o decréscimo da temperatura é definido um fator de resfriamento (veja abaixo), as condições de transição de fase são verificadas e, se necessário, um novo protótipo é criado. A cada passo, os cálculos de probabilidades e as posições dos protótipos são atualizadas. No final do processo, quando é atingido o valor mínimo de temperatura, os protótipos são fixos e os pontos Apex são definidos a um protótipo. São aceitos protótipos com mais de dois pontos Apex associados.
- Para cada protótipo, o conjunto de pontos Apex é substituído pela trajetória. Cada protótipo torna-se uma semente de vértice;
- Todas as sementes de vértices são ajustadas.

Desta forma, o algoritmo DA reconstrói todos os vértices primários e secundários ao mesmo tempo, sem conhecimento prévio do número de vértices que devem ser encontrados.

Resultados de simulações deste método para 4000 eventos  $b\bar{b}$  sem empilhamento, mostra uma eficiência de  $92,75 \pm 0,42\%$  para vértices primários.

### 3.1.6 Localização pelo Método dos Multiplicadores de Lagrange

Considera-se o conjunto de medidas  $y = (y_1, \dots, y_n)$ , que será a soma de quantidades verdadeiras  $\eta$  (desconhecidas) e de erros de medidas  $\epsilon = (\epsilon_1, \dots, \epsilon_n)$ :

$$y_j = \eta_j + \epsilon_j \quad j = 1, \dots, n \quad (3.6)$$

A matriz de covariância das medidas  $y$  serão escritas como  $C_y$  e cada  $\epsilon_j$  será considerado uma variável randômica, normalizada em torno de zero, com variância  $(C_y)_{jj} = \sigma_j^2$ . A função de densidade de probabilidade das variáveis  $y$  pode ser escrita da seguinte forma:

$$\phi(y) = \frac{1}{(2\pi)^{\frac{N}{2}} \text{sqrt}(\det(G_y))} \exp \left[ -\frac{1}{2} (y - \eta)^t G_y (y - \eta) \right] \approx \exp \left[ -\frac{1}{2} \epsilon^t G_y \epsilon \right] \quad (3.7)$$

onde  $G_y = C_y^{-1}$  é a inversa da matriz de covariância, e  $\epsilon^t$  é a transposta de  $\epsilon$ . A função de verossimilhança logarítmica correspondente é, desconsiderando fatores constantes,  $l = -\epsilon^t G_y \epsilon$ , e tem um máximo em  $\epsilon = 0$ . Isso significa que se nenhuma informação extra for dada ao sistema, a melhor estimativa para as quantidades  $\eta$  é a própria medida de  $y$ .

Assume-se que as equações de vínculo são representadas por:

$$f_k(\eta) = 0 \quad j = k, \dots, q \quad (3.8)$$

onde  $q$  é o número de graus de liberdade do sistema e  $k$  é o número de equações de vínculo do sistema.

Em geral, as funções  $f_k$  não são lineares, mas será assumido que estas funções podem ser bem descritas por uma expansão de Taylor de primeira ordem na vizinhança de  $\eta_0$ , que representa uma primeira aproximação para valores verdadeiros de  $\eta$ :

$$f_k(\eta) \approx f_k(\eta_0) + \left( \frac{\partial f_k}{\partial \eta} \right)_{\eta_0} (\eta - \eta_0) = 0 \quad (3.9)$$

A equação 3.9 pode ser escrita de uma maneira compacta em forma matricial:

$$B\delta + c = 0 \quad (3.10)$$

com a seguinte notação:

$$B_{kl} = \left( \frac{\partial f_k}{\partial \eta_l} \right)_{\eta_0} \quad c_k = f_k(\eta_0) \quad \delta = \eta - \eta_0 \quad (3.11)$$

Esta informação extra é incluída na função de densidade de probabilidade pela adição de um fator gaussiano:

$$\phi(y) \approx \exp \left( -\frac{1}{2} \delta^t G_y \delta \right) \exp [-\mu^t (B\delta + c)] \quad (3.12)$$

onde o vetor  $\mu = (\mu + 1, \dots, \mu_q)$  contém os multiplicadores de Lagrange. Os valores medidos  $\eta_0 = y$  serão considerados uma primeira aproximação para  $\eta$ .

A ferramenta de reconstrução de vértices por multiplicadores de Lagrange faz um ajuste de vértice com vínculo geométrico e com vínculo de massa invariante. Somente partículas com largura menor que  $1 \text{ MeV}/c^2$  serão consideradas como ressonância[6].

O vínculo geométrico é a condição de que todas as partículas devem ter um ponto comum no espaço. O vínculo de massa invariante, por outro lado, é a condição de que todas as partículas filhas, juntas, deve dar a massa invariante da partícula mãe.

Cada partícula filha é representada por um traço. O vínculo geométrico exige que as partículas filhas tenham um ponto comum no espaço. Neste ponto, as coordenadas  $(x, y, z)$  para todas as partículas filhas devem ser iguais separadamente.

O número de graus de liberdade do sistema é igual ao número de equações de vínculo do sistema. Para um número grande de partículas filhas, o procedimento utilizado nessa ferramenta de Lagrange exige a inversão de matrizes grandes e a obtenção de várias derivadas parciais. Portanto, outros métodos devem ser utilizados.

A ferramenta dos multiplicadores de Lagrange está implementada em C++ e consiste na ferramenta utilizada pelo sistema de reconstrução de vértices secundários de colisão do LHCb.

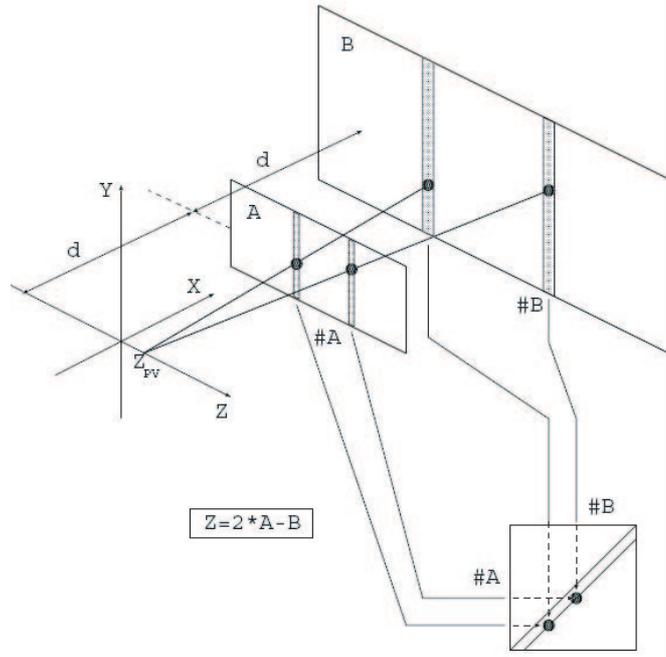


Figura 3.9: Ilustração do algoritmo de ajuste linear. Extraído de [6].

### 3.1.7 Localização pelo Método de Ajuste Linear (*Least Square*)

O método de ajuste linear realiza uma relação matemática simples entre as coordenadas dos pontos de interseção da trajetória da partícula, os dois planos de detecção dos detectores de vértice e a coordenada do vértice primário ( $z_{pv}$ ) ao longo do eixo do feixe (eixo  $z$ ). Os detectores de traço estão posicionados de forma transversal ao eixo do feixe de forma que o segundo plano de detecção está no dobro da distância do primeiro, como ilustra a Figura 3.9. O ponto de colisão  $z_{pv}$  é obtido pela equação:

$$z_{pv} = 2z_A^i - z_B^i \quad (3.13)$$

onde  $(z_A^i; z_B^i)$  correspondem a posição no eixo  $z$  dos planos  $A$  e  $B$  da  $i$ -ésima trajetória.

A equação 3.13 está correta para todo  $z_{pv}$  com coordenadas transversas  $x_{pv} = y_{pv} = 0$  e corresponde a uma boa aproximação para pequenas variações no eixo do feixe, como é o caso do LHC.

Para um evento a posição  $z_{pv}$  é fixa ( $z_{pv}^*$ ). Desta forma, em um gráfico de

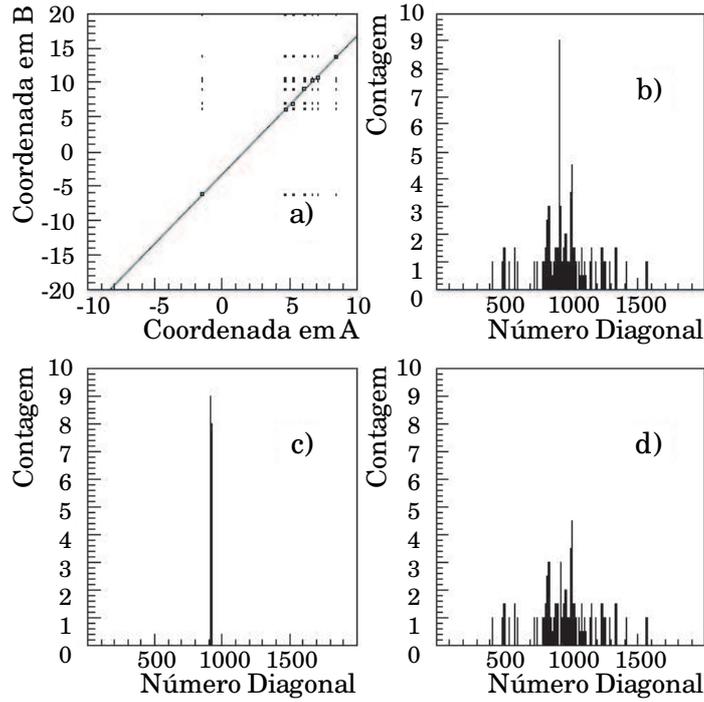


Figura 3.10: Correlação entre as coordenadas  $Z_A^i$  e  $Z_B^i$  para trajetórias oriundas do vértice primário. (a) Medida de todas as combinações das coordenadas em  $A$  e  $B$ . Note que a escala no plano  $B$  é duas vezes maior que no plano  $A$ . As combinações verdadeiras formam uma linha reta. (b) Contagem da combinação de todas as diagonais. (c) somente as combinações verdadeiras são contadas. (d) Contagem das combinações erradas. Extraído de [6].

$z_A^i \times z_B^i$  todas as trajetórias oriundas de  $z_{pv}$  formarão uma linha reta ( $z_B^i = 2z_A^i - z_{pv}^*$ ). No entanto, se o número de pontos no plano  $B$  for duas vezes maior que no plano  $A$ , a relação entre o número de pontos entre  $A \times B$  será uma diagonal (Figura 3.10a).

Esta abordagem pode ser implementada em hardware como uma matriz de coincidência onde os pontos do plano  $A$  são combinados com os do plano  $B$ . Depois são contadas as combinações formadas ao longo das diagonais. Na distribuição das contagens um pico é esperado para a diagonal que corresponde ao  $z_{pv}$  e um ruído de fundo que corresponde as combinações se espalham ao longo do eixo do feixe.

A decisão de quando o  $z_{pv}$  é encontrado está baseada na informação de ambos os lados dos detectores de traço. Em um passo inicial, os pontos de cada lado são combinados separadamente. Depois é analisada a soma dos dois histogramas. Caso seja encontrado somente um pico na distribuição das contagens das diagonais esta

posição é considerada como  $z_{pv}$  encontrado. Caso contrário a decisão não é tomada. A posição do  $z_{pv}$  candidato é:

$$z_{pv} = \frac{\sum_{i=-1}^1 z_{pv}^i h_i}{\sum_{i=-1}^1 h_i} \quad (3.14)$$

onde  $z_{pv}^i$  é a coordenada que corresponde ao pico ( $i = 0$ ) e os canais vizinhos ( $i = -1, 1$ ),  $h_i$  é a contagem em cada canal do histograma.

Este sistema encontra-se implementado em hardware no sistema de filtragem do LHCb.

## 3.2 Sistemas de Localização de Vértices Primários do ATLAS

Os sistemas de localização de vértices primários do ATLAS se encontram em funcionamento no segundo nível de filtragem. Uma das principais finalidades consiste em utilizar o vértice primário para reduzir o tamanho da região de interesse, reduzindo assim o volume de dados na reconstrução *online* de trajetórias.

Atualmente o ATLAS utiliza dois sistemas distintos de reconstrução do vértice primário, que são: o sistema de localização por ajuste linear de retas (*ZFinder*) e o sistema de localização através do calorímetro eletromagnético. Ambos sistemas são utilizados pelo LVL2 na primeira etapa de reconstrução de trajetórias de partículas.

As Seções seguintes abordam os detalhes de funcionamento de ambos os métodos. É importante ressaltar que somente o método *ZFinder* foi utilizado para comparação com o método de reconstrução por transformada de Hough desenvolvido nessa Tese. Isto se deve ao fato deste método estar em funcionamento há mais de uma década, enquanto o de localização através do calorímetro eletromagnético ter entrado em funcionamento no final de 2005.

### 3.2.1 Localização por Ajuste Linear de Retas (*ZFinder*)

A aplicação de localização por ajuste linear de retas na direção  $\rho \times z$ , denominada *ZFinder*, é desenvolvida pelo grupo PESA. De forma diferente do método descrito na seção anterior, o vértice de colisão é encontrado antes do processo de

reconstrução. O objetivo é reduzir o tempo de processamento e melhorar a performance dos sistemas de reconhecimento de padrões nos detectores de traço do ATLAS.

O algoritmo está baseado na relação aproximadamente linear entre  $z$  e  $\rho$  das trajetórias de partículas em campos magnéticos uniformes paralelos ao eixo  $z$  (do feixe)[49].

A trajetória de uma partícula carregada dentro de um campo magnético uniforme é uma hélice. Em coordenadas cilíndricas, com o eixo  $z$  paralelo ao campo magnético, a hélice passa pelo vértice com coordenadas  $(0, 0, z_v)$  e pode ser descrita pela equação 3.15

$$\rho = Gp_t \sin(\phi - \phi_0) \cong Gp_T(\phi - \phi_0) = G'(\phi - \phi_0) \quad (3.15)$$

$$z - z_v = Gp_z(\phi - \phi_0) \quad (3.16)$$

onde  $p_T$  e  $p_z$  são os momentos transversos e longitudinal,  $G$  é uma constante conhecida que depende do campo magnético e  $G' = Gp_T$ . A partir destas equações uma terceira pode ser obtida, equação 3.17,

$$z - z_v \cong \frac{p_z}{p_T} \rho = C\rho \quad (3.17)$$

que é linear em uma boa aproximação. Desta forma, se  $(z_1, \rho_1)$  e  $(z_2, \rho_2)$  são coordenadas de dois pontos ao longo da trajetória da partícula (com  $\rho_1 \neq \rho_2$ ),  $z_v$  pode ser calculado por uma simples extrapolação linear utilizando a equação 3.18.

$$z_v = \frac{z_2\rho_1 - z_1\rho_2}{\rho_1 - \rho_2} \quad (3.18)$$

Desta forma, o *ZFinder* consiste no primeiro algoritmo a utilizar as coordenadas cilíndricas para aproximar a hélice observada em  $xyz$  para uma reta em  $\phi\rho z$ . Esta mudança de sistema de coordenadas permitiu uma simplificação no cálculo do ajuste.

Nesse trabalho, o algoritmo *ZFinder* foi estudado de forma bastante detalhada. O capítulo 7 apresenta uma descrição da implementação deste algoritmo, e os resultados quando aplicado com eventos de baixa e alta luminosidade. É importante

ressaltar que esses resultados serão utilizados como parâmetro de comparação para os algoritmos de localização por transformada de Hough desenvolvidos.

### 3.2.2 Localização através do Calorímetro Eletromagnético

Os calorímetros vêm possuindo um papel de crescente importância nos experimentos com colisionadores. Estes detetores medem a energia das partículas incidentes por meio de sua absorção total. O perfil de deposição da energia no detetor depende do tipo de partícula que com ele interage. Com isto, os calorímetros são usados em sistemas *online* de validação de eventos, aproveitando da sua alta eficiência de discriminação e de sua velocidade de processamento.

O calorímetro eletromagnético possui sensibilidade para detectar partículas de elétrons, gamas ( $\gamma$ ) e taus ( $\tau$ ). A região de interesse deste calorímetro, denominada EmTau RoI, é determinada inicialmente pelo primeiro estágio de filtragem do ATLAS. O objetivo deste sistema de localização consiste em refinar a EmTau RoI estipulada no LVL1 dentro do primeiro estágio do segundo nível de filtragem (LVL2). No LVL2 toda a granularidade do calorímetro eletromagnético encontra-se disponível. Desta forma, o LVL2 pode determinar regiões com precisão mais fina que no LVL1[55].

A Figura 3.11 ilustra a região de interesse na projeção  $\rho \times z$ . No LVL1 esta região de interesse possui as dimensões fixas em  $(\Delta\eta \times \Delta\phi) = (0,2 \times 0,2)$  e na região da linha de feixe se estende em  $z = 16,8\text{cm}$ , levando em consideração toda a variação possível de localização do vértice primário de colisão.

Esse método propõe utilizar a informação do calorímetro eletromagnético para estimar a posição do vértice primário e possibilitar a redução significativa da região de interesse no LVL2. A Figura 3.12 apresenta uma ilustração da região determinada por esse método. No eixo do feixe uma variação de  $\pm 3\sigma$  é dada na vizinhança do vértice encontrado. Desta forma, a região de interesse no eixo do feixe não representa toda a região possível de colisão. Os parâmetros  $\eta_F$  e  $\eta_B$  são considerados iguais nesse estudo.

O método de localização do vértice primário pela informação do calorímetro utiliza a aproximação que trajetórias com alto momento transversal são aproximadamente lineares no plano  $\rho \times z$ . Desta forma, a posição  $z_{vtx}$  do vértice primário pode

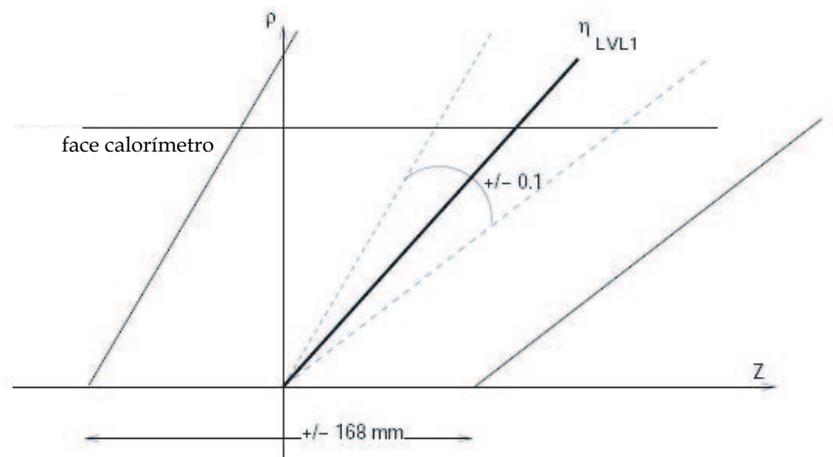


Figura 3.11: Exemplo de uma região de interesse do LVL2 no plano  $\rho \times z$ .

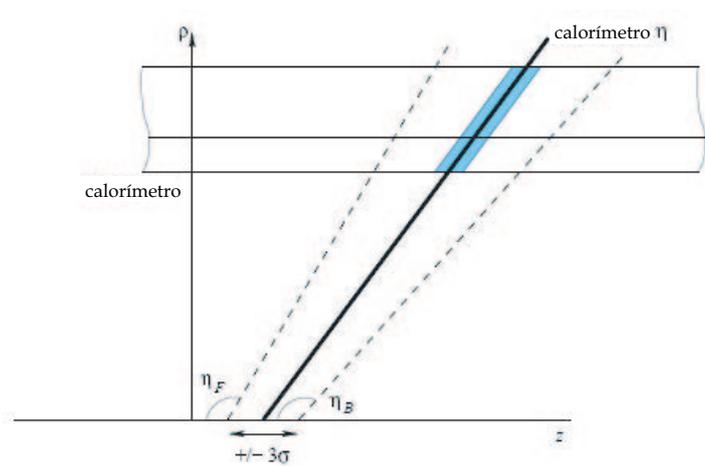


Figura 3.12: Região de interesse proposta no plano  $\rho \times z$ .

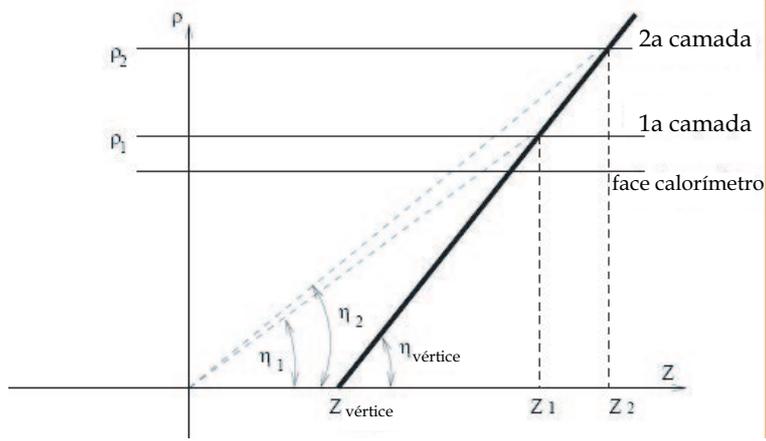


Figura 3.13: Determinação dos valores das Equações 3.19 e 3.20.

ser estimada por uma extrapolação linear. Dada as coordenadas  $(\rho, z)$  de duas camadas do calorímetro, como ilustra a Figura 3.13, a posição  $z_{vtx}$  é calculada pela equação

$$\frac{z_1}{\rho_1} = \sinh \eta_1 \quad (3.19)$$

$$\frac{z_2}{\rho_2} = \sinh \eta_2 \quad (3.20)$$

$$z_{vtx} = \frac{z_1 \rho_2 - z_2 \rho_1}{\rho_2 - \rho_1} \quad (3.21)$$

Esse sistema de localização foi testado com amostras de elétrons individuais a baixa e alta luminosidade. O teste de eficiência e tempo de processamento foi realizado da seguinte forma: os eventos de alta e baixa luminosidade eram processados pelo algoritmo *ZFinder* com a região de interesse do LVL1. Em seguida, era calculada a nova região de interesse por esse método de localização utilizando informações dos calorímetros. O algoritmo *ZFinder* era executado nesta nova região de interesse e o resultado comparado com a execução utilizando a região de interesse completa. Foi considerado como acerto quando o vértice primário localizado pelo *ZFinder* se encontrava em uma vizinhança de 2 mm do vértice simulado ou  $z_{true}$ , conforme na Equação 3.22.

$$|z_{vtx} - z_{true}| \leq 2mm \quad (3.22)$$

A Tabela 3.1 apresenta a eficiência do algoritmo *ZFinder* para regiões de interesse reduzidas e originais do LVL1. É possível observar que houve um au-

mento na eficiência do algoritmo *ZFinder* com a região de interesse reduzida por esse método, principalmente nos eventos de alta luminosidade, onde a eficiência passou de  $89,58 \pm 0,31\%$  para  $95,01 \pm 0,22\%$ . A Tabela 3.2 apresenta o tempo de processamento do algoritmo *ZFinder* com a região original e com a reduzida. Houve uma diminuição significativa no tempo de processamento para amostras com alta luminosidade, onde este passou de 0,81ms para 0,34ms. Todos os tempos de processamento foram normalizados de forma linear para uma máquina de 4GHz.

	Baixa Luminosidade		Alta Luminosidade	
	RoI original	RoI reduzida	RoI original	RoI reduzida
Barris	$96,11 \pm 0,38$	$97,36 \pm 0,31$	$90,12 \pm 0,42$	$96,69 \pm 0,25$
Transição	$95,00 \pm 1,54$	$94,50 \pm 1,61$	$92,62 \pm 0,97$	$95,63 \pm 0,76$
<i>End-cap</i>	$93,96 \pm 0,55$	$95,35 \pm 0,49$	$88,24 \pm 0,53$	$92,59 \pm 0,43$
Total	$95,21 \pm 0,31$	$96,44 \pm 0,27$	$89,58 \pm 0,31$	$95,01 \pm 0,22$

Tabela 3.1: Eficiência do algoritmo *ZFinder* em eventos de elétrons individuais com baixa e alta luminosidade para regiões de interesse reduzidas pelo método de localização através da informação do calorímetro e para regiões de interesse originais obtidas do LVL1.

	Baixa Luminosidade		Alta Luminosidade	
	tempo <i>ZFinder</i>	Num SP	tempo <i>ZFinder</i>	Num SP
RoI original	0,32ms	43,06	0,81ms	156,2
RoI reduzida	0,23ms	18,12	0,34ms	44,07
% diferença	-28,13%	-57,92%	-58,02%	-71,79%

Tabela 3.2: Tempos de processamento do algoritmo *ZFinder* em eventos de elétrons individuais com baixa e alta luminosidade para regiões de interesse reduzidas pelo método de localização através da informação do calorímetro e para regiões de interesse originais obtidas do LVL1.

## 3.3 Sistemas de Reconstrução de Trajetórias de Partículas

Sistemas de reconstrução de trajetórias de partículas são indispensáveis para os detectores de traço. Através deles é possível reconhecer e analisar assinaturas de partículas deixadas nos detectores. Normalmente, os sistemas de reconhecimento de trajetórias possuem elevada complexidade computacional e, desta forma, são utilizados em ambientes de processamento *offline*.

Os sistemas de reconstrução de trajetórias utilizados no ambiente *online* possuem restrições de tempo de execução e, portanto, possuem menos resolução no ajuste da trajetória. Alguns métodos de reconstrução são adaptados para localização do vértice primário de colisão. A seção 3.1.2 apresentou a utilização da filtragem de Kalman como método de localização de vértices. No entanto, em [43] e [50] a filtragem de Kalman é utilizada para reconhecimento de padrões e reconstrução de trajetórias.

Esta seção apresenta alguns métodos de reconstrução de trajetórias utilizados em experimentos do CERN. É importante ressaltar que eles não representam o universo dos métodos já empregados em problemas de reconstrução de trajetória de partículas.

### 3.3.1 Reconstrução de Trajetórias por Filtragem de Kalman

O método de reconstrução de trajetórias por filtragem de Kalman consiste em um dos métodos mais utilizados e difundidos no ambiente de Física de Altas Energias. O método de filtragem de Kalman [56, 57] permite realizar tarefas de reconhecimento de padrões e ajuste de trajetórias de forma simultânea[58].

O formalismo de Kalman realiza atualizações da informação das trajetórias candidatas durante o processo de localização de trajetórias e fornece informações precisas do ponto seguinte a ser localizado. Desta forma, quando terminado o processo de reconhecimento de padrões, o ajuste dos parâmetros da trajetória é obtido sem a realização de mais cálculos. Ao mesmo tempo, as trajetórias ajustadas estão preparadas para serem casadas com outras trajetórias dos detectores vizinhos.

O método de reconstrução por filtragem de Kalman utilizado por Quian e

Billoir em [3] utiliza a projeção no plano  $x \times y$  dos detectores de traço. Os parâmetros de trajetória são:  $P_n = (\phi, \sin(\beta), 1/R_{tr})$ , representado na Figura 3.14, em um raio fixo  $r_n$  onde  $n$  representa a localização do  $n$ -ésimo plano de medida,  $\phi$  o ângulo azimutal,  $\beta$  o ângulo entre a direção  $\phi$  e a tangente da trajetória em  $r_n$ ,  $R_{tr}$  é o raio da curvatura da trajetória que é proporcional ao momento transverso  $p_t$ .

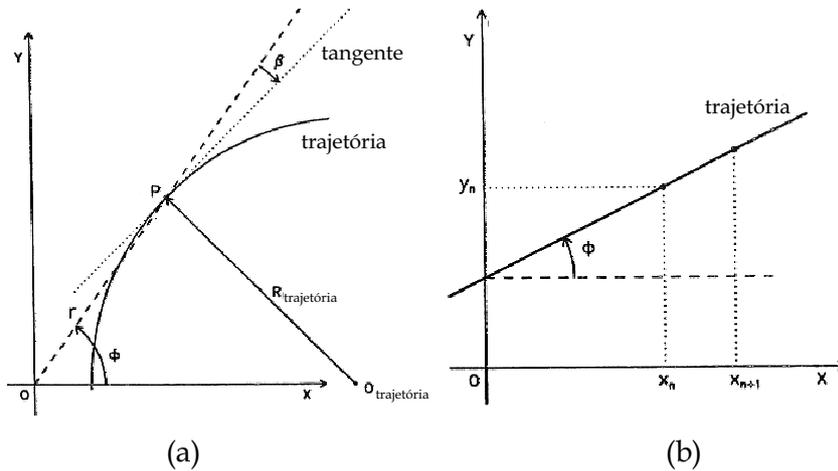


Figura 3.14: Representação dos parâmetros da trajetória nos eixos  $x$  e  $y$ .

O processo de ajuste de trajetórias considera como primeiros candidatos todos os pontos da camada mais externa do detector, denominados de pontos de partida. Um ponto de partida mais um ponto candidato da camada adjacente dentro de uma janela de alcance forma os primeiros segmentos de possíveis trajetórias. O tamanho da janela depende do detector e é ajustado de acordo com o momento transverso  $p_t$ . Em seguida, estes segmentos mais os pontos da camada inferior formam uma cadeia de três pontos. Estes são ajustados pela equação de um círculo e os valores de  $P$  são calculados. Este processo se repete até a camada mais interna ser alcançada.

### 3.3.2 Reconstrução de Trajetórias por Braços Elásticos

O método de braços elásticos baseia-se em modelos de deformação que são resolvidos por algoritmos de resfriamento determinístico (*deterministic annealing*). A tarefa de reconstrução de trajetórias consiste em um problema de otimização combinatória. As técnicas de redes neurais artificiais, ou suas variações, mostram ser eficientes aproximadores de soluções para problemas complexos de otimização. A abordagem neural foi testada em problemas de reconstrução de trajetórias com

resultados encorajadores [59, 60].

A estratégia do método de braços elásticos consiste em casar eventos de colisões observados com modelos de parametrização simples, que contenham conhecimento *a priori* sobre possíveis trajetórias. O problema de reconstrução de trajetórias consiste em ajustar curvas suaves em um conjunto de pontos, correspondentes a localização da resposta dos sensores. Neste método é assumido que as possíveis trajetórias são círculos que passam pela origem, ou ponto de colisão.

A transformada de Hough[61] é utilizada para prover as condições iniciais para os modelos de parametrização, e especificar o número de modelos. A transformada de Hough consiste em técnicas de histogramação e foram utilizadas para reconstrução de trajetórias por [62].

A abordagem por braços elásticos, ou modelos deformáveis, consiste em utilizar as redes neurais artificiais e a transformada de Hough Local para localizar trajetórias de eventos. O algoritmo busca minimizar uma função de energia  $E$  que contém parâmetros de um círculo. Os parâmetros iniciais da função de energia são fornecidos pela transformada de Hough do evento. Na temperatura inicial, uma série de modelos gaussianos é localizada de acordo com os parâmetros da transformada de Hough. Os modelos consistem em distribuições gaussianas centradas nos valores dos braços e a largura é dada pela temperatura. A Figura 3.15 apresenta uma ilustração deste processo. Quando a temperatura é baixada, os diferentes braços são atraídos pelos mais próximos formando um novo braço.

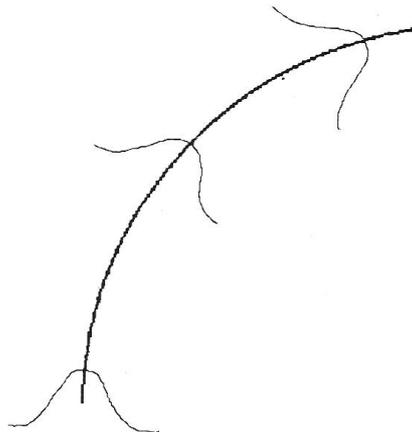


Figura 3.15: Ilustração do processo de distribuição gaussiana nos braços (ou trajetórias).

Este método encontra-se detalhado em [63, 64, 65]. A Figura 3.16 apresenta o resultado obtido para um evento simulado do experimento DELPHI no CERN.

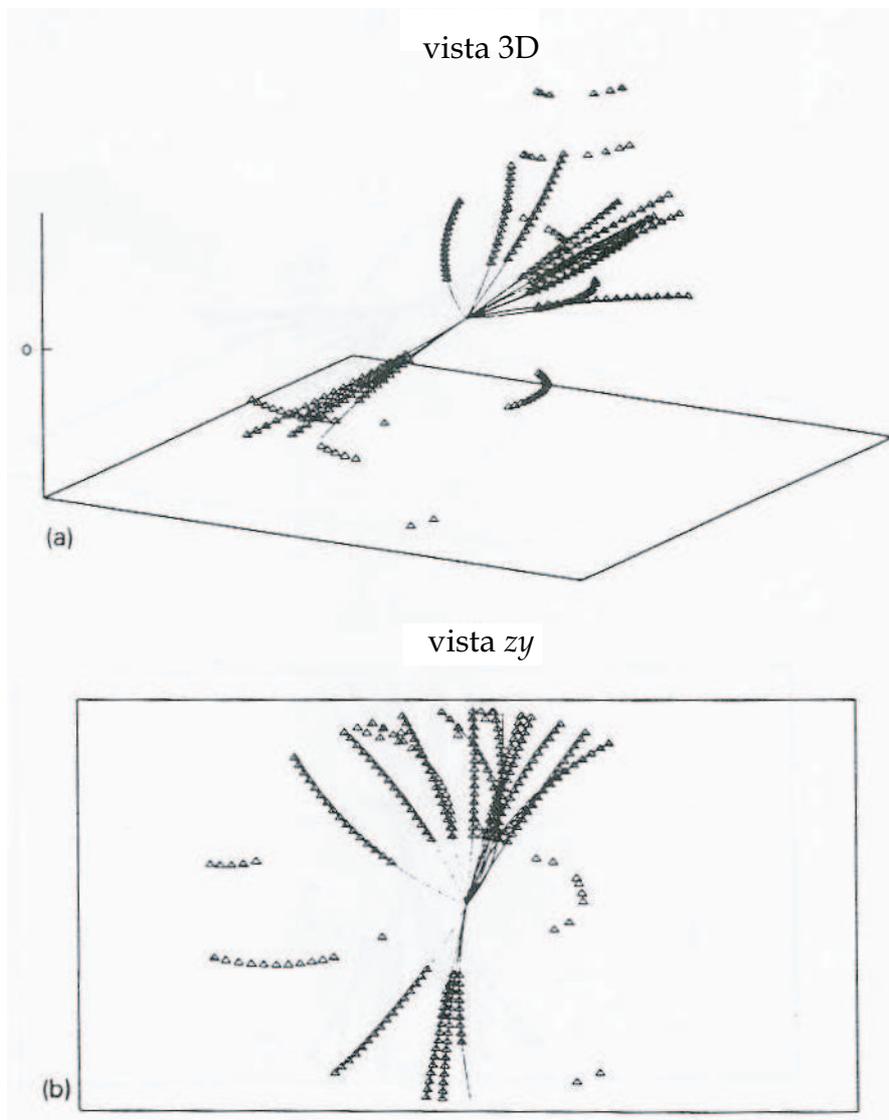


Figura 3.16: Resultado do algoritmo de braços elásticos em uma simulação de evento.

### 3.3.3 Reconstrução de Trajetórias por Transformada de Hough

A reconstrução de trajetórias de partículas por transformada de Hough foi abordada em [47] utilizando imagens de colisões do detector SFT (*Scintillating Fibre Tracker*) do experimento L3 no CERN. O objetivo deste trabalho consistiu em estudar a aplicação da transformada de Hough na detecção de trajetórias na projeção  $x \times y$  do STF. A Figura 3.17 apresenta a imagem de um evento típico deste detector. O algoritmo não possuía informações *a priori* de quantas trajetórias estavam

representadas nas imagens.

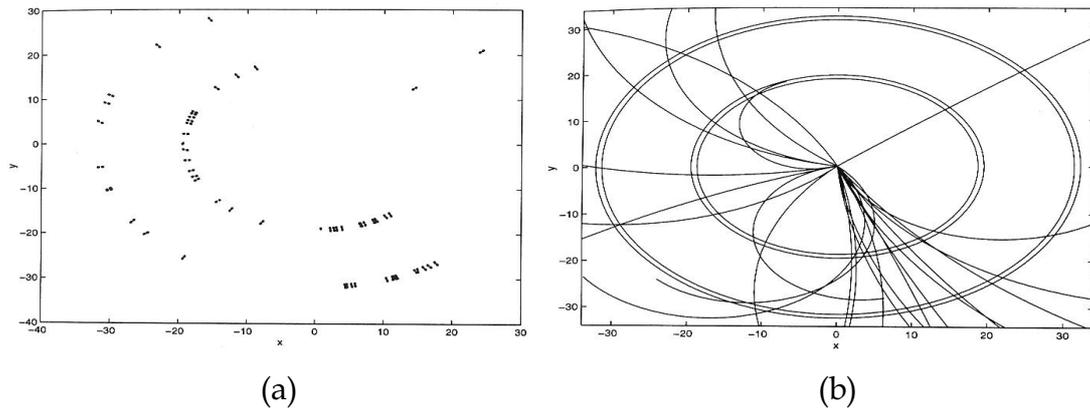


Figura 3.17: Exemplo de um evento de colisão. (a) imagem  $x \times y$  do detector de traço SFT. (b) simulação da imagem em (a).

No evento da Figura 3.17(a) estão “escondidas” 25 hélices ou trajetórias. A Figura 3.17(b) representa as trajetórias simuladas que deram origem ao evento. O trabalho de reconstrução de trajetórias por Hough utilizou um círculo como função de ajuste dos pontos na imagem.

A transformada de Hough consiste em um método matemático bastante utilizado na detecção de padrões presentes em imagens binárias que contenham ruído ou dados incompletos[66, 67, 68]. Tais características encontram-se nas imagens dos detectores de traço, motivando o desenvolvimento da ferramenta de reconstrução por transformada de Hough.

A desvantagem dos métodos de Hough consiste na elevada carga computacional por ela requerida, principalmente quando implementada em sua forma original ou padrão. Esta desvantagem motivou o teste de diferentes abordagens da transformada de Hough como a Local, Adaptativa e Randomizada. Este trabalho realizou a comparação de eficiência e tempo de execução destas variantes.

O Capítulo 4 apresenta com mais detalhes os conceitos de transformada de Hough e suas variações. A Tabela 3.3 apresenta o resultado obtido neste estudo para eventos como o ilustrado da Figura 3.18.

Algoritmo	Eficiência	Tempo Execução (s)
Padrão	78 %	218,41
Local	84,2%	3,09
Randomizada básica	88,5%	2,20

Tabela 3.3: Eficiências obtidas para os diversos algoritmos testados neste trabalho.

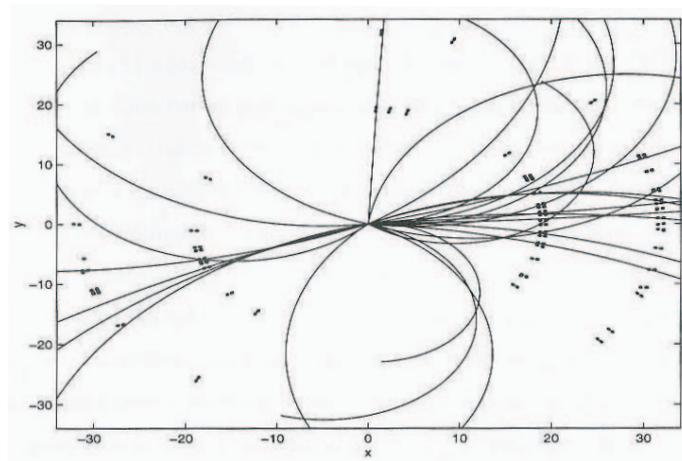


Figura 3.18: Exemplo de um resultado da transformada de Hough em um evento de colisão do SFT.

# Capítulo 4

## A Transformada de Hough

Este capítulo apresenta a transformada de Hough, analisando suas variações e sua aplicação na reconstrução do vértice de colisão. Serão abordados: as características, as vantagens, as desvantagens e os algoritmos.

A transformada de Hough foi inicialmente proposta em 1962 por Paul V. C. Hough, em artigo intitulado “*A method and means for recognizing complex patterns*” [69]. Sua idéia inicial era utilizar a transformada para a detecção de padrões complexos, a partir de pontos em uma imagem binária. Mais especificamente, Hough testou a aplicação de sua transformada na detecção de curvas em determinadas fotografias. Seu trabalho, porém, só veio a despertar maior atenção daqueles que estavam à frente nas pesquisas com processamento de imagem quando publicado em 1969, por A. Rosenfeld, com o artigo “*Picture Processing by Computer*” [70], e ganhou ainda mais importância após a divulgação do trabalho de R. D. Duda e P. E. Hart, “*Use of the Hough transform to detect lines and curves in pictures*” [70].

A transformada de Hough foi reconhecida, naquela época, como uma técnica quase única na análise de imagens com ruído e dados incompletos; porém, sua utilização foi bastante lenta devido, principalmente, ao elevado custo e complexidade computacional requeridos para a sua implementação. Mesmo com relação aos aspectos teóricos, havia a falta de uma maior compreensão de suas propriedades, o que deixava receosos os pesquisadores da época. Entretanto, considerável progresso tem sido feito nos últimos anos em relação a implementações mais eficazes, e diversas novas variações da transformada permitem aplicações mais específicas, que variam de acordo com o problema em questão. A descoberta da relação da transformada

de Hough com outros métodos de reconstrução e detecção de padrões de imagens tem incentivado a pesquisa na área de Física de partículas, fazendo com que novas aplicações surjam para diversos problemas.

## 4.1 Transformada de Hough Padrão - THP

A transformada de Hough (TH) é o mapeamento do  $\mathbb{R}^2$  em um espaço de retas ou de funções senoidais. De forma semelhante, podemos definir a TH como um mapeamento de um determinado espaço de dados para o respectivo espaço de parâmetros. Esta seção apresenta duas formulações para a Transformada de Hough Padrão aplicada a detecção de retas em um conjunto de pontos no  $\mathbb{R}^2$ . A primeira consiste na utilização do mapeamento em retas e a segunda em funções senoidais. Início pela determinação do mapeamento da equação analítica da reta.

### 4.1.1 Detecção de Retas

Uma “linha” reta para o algoritmo de Hough é representada por um conjunto colinear de pontos. A qualidade da “linha” é julgada pelo número de pontos que lhe pertence. Caso um conjunto de pontos no  $\mathbb{R}^2$  com coordenadas  $(x, y)$  representem uma reta, sua representação é dada pela equação:

$$\vec{y} = m.\vec{x} + c \quad (4.1)$$

onde  $m$  representa o coeficiente angular e  $c$  o coeficiente linear da reta. O método de “força bruta” para encontrar linhas retas em um conjunto discreto de  $N$  pontos espaciais consiste em examinar todas as  $\frac{N(N-1)}{2}$  possibilidades de linhas retas entre os pontos. Para cada  $\frac{N(N-1)}{2}$  possíveis linhas,  $N - 2$  testes de colinearidade devem ser realizados. Desta forma, este método possui uma complexidade computacional de  $N^3$ .

De modo a iniciar a descrição do algoritmo de Hough, primeiro define-se a Transformada de Hough Padrão (THP) e, em seguida, examinam-se algumas de suas propriedades. A THP é o mapeamento  $h$  de  $\mathbb{R}^2$  em funções analíticas de retas (4.2).

$$h : (x, y) \rightarrow c = -mx + y \quad (4.2)$$

Para que a THP possa ser utilizada para encontrar linhas retas em um conjunto discreto de pontos, é feita a seguinte consideração:

Qualquer linha reta  $l_o$  no plano- $xy$  corresponde a um ponto  $(m_o, c_o)$  no plano- $mc$ , onde  $m_o, c_o \in \mathbb{R}$ .

A Figura 4.1 apresenta um exemplo. Dado os pontos  $(1,7)$ ,  $(3,5)$ ,  $(5,3)$  e  $(6,2)$  no plano- $xy$ . Estes se encontram ao longo da reta  $l_o$  com  $m = -1$  e  $c = 8$ .

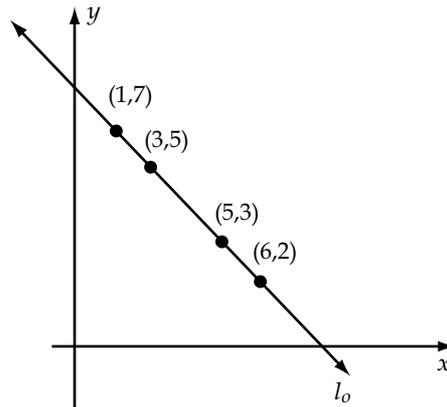


Figura 4.1: Exemplo de aplicação da THP em um conjunto de quatro pontos que definem uma reta de coeficiente angular igual a -1 e linear igual a 8.

A Transformada de Hough Padrão mapeia os pontos indicados em funções, como na equação 4.2.

$$h : (1, 7) \rightarrow c = -1m + 7$$

$$h : (3, 5) \rightarrow c = -3m + 5$$

$$h : (5, 3) \rightarrow c = -5m + 3$$

$$h : (6, 2) \rightarrow c = -6m + 2$$

O gráfico destas retas podem ser observados da Figura 4.2. Note como as quatro retas se interceptam no ponto  $m = -1$  e  $c = 8$ .

A abordagem da THP pela equação 4.2 possui uma instabilidade, pois tanto a inclinação quanto a interseção em  $y$  tendem a infinito quando a reta se aproxima da vertical. Esta instabilidade não é observada na abordagem por funções senoidais.

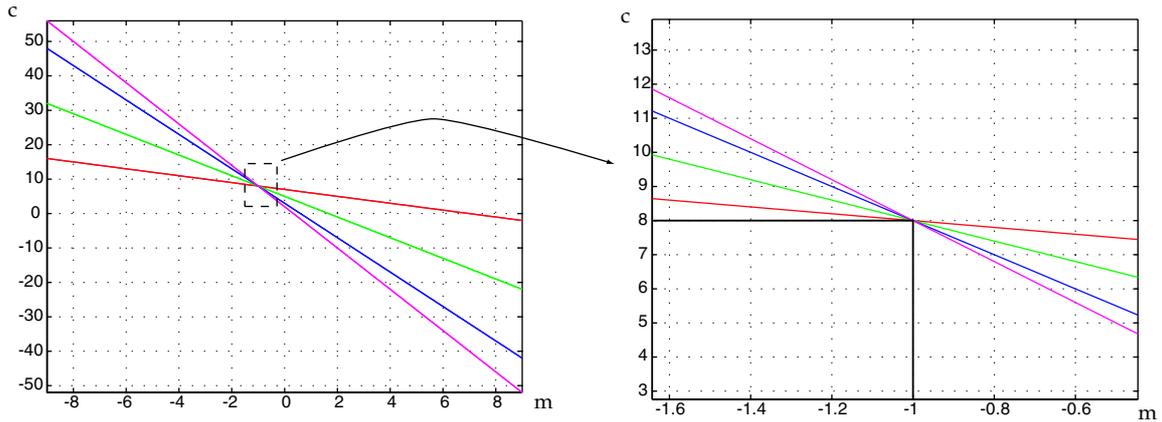


Figura 4.2: Exemplo do espaço  $m, c$ . Os quatro pontos do plano- $xy$  tornaram-se retas no plano- $mc$ . O ponto de interseção indica o coeficiente angular e linear da reta no plano- $xy$ .

### 4.1.2 Descrição de Retas por Funções Senoidais

A descrição da THP por funções senoidais utiliza o mapeamento de funções senoidais através da equação (4.3).

$$h : (x, y) \rightarrow \rho = x \cos(\theta) + y \sin(\theta) \quad (4.3)$$

onde  $\rho$  e  $\theta$  estão representados na Figura 4.3

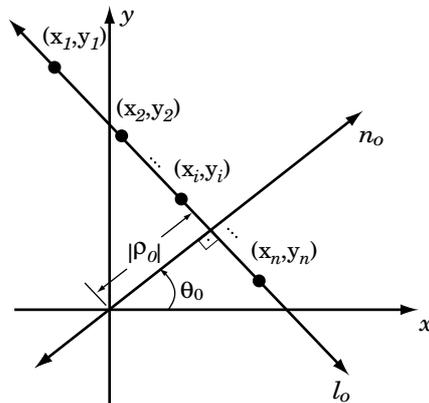


Figura 4.3: Representação de  $\rho$  e  $\theta$ .

De modo a utilizar a THP para encontrar linhas retas em um conjunto discreto de pontos, temos que fazer as seguintes considerações:

Qualquer linha reta  $l_o$  no plano- $xy$  corresponde a um ponto  $(\rho_o, \theta_o)$  no plano- $\rho\theta$ , onde  $\theta_o \in [0, \pi)$  e  $\rho_o \in \mathbb{R}$ .

Seja  $n_o$  a reta normal à  $l_o$  que passa pela origem do plano- $xy$ . O ângulo que  $n_o$  faz com o eixo positivo de  $x$  é  $\theta_o$ . A distância de  $(0, 0)$  à  $l_o$  ao longo de  $n_o$  é  $|\rho_o|$ . A Figura 4.3 ilustra a relação entre  $l_o$ ,  $n_o$ ,  $\theta_o$  e  $\rho_o$ . Note que o eixo- $x$  na Figura 4.4 corresponde ao ponto  $(0, 0)$ , enquanto que o eixo- $y$  corresponde ao ponto  $(0, \pi/2)$ .

Suponha que  $(x_i, y_i)$ ,  $1 \leq i \leq n$ , são pontos do plano- $xy$  que estão dispostos ao longo da linha reta  $l_o$ , como na Figura 4.3. A linha  $l_o$  possui a representação  $(\rho_o, \theta_o)$  no plano- $\rho\theta$ . A THP mapeia cada ponto  $(x_i, y_i)$  na função senoidal  $\rho = x_i \cos(\theta) + y_i \sin(\theta)$  no plano- $\rho\theta$ . Esta transformada baseia-se na propriedade que cada curva  $\rho = x_i \cos(\theta) + y_i \sin(\theta)$  possui um ponto em comum de interseção, denominado  $(\rho_o, \theta_o)$ . De forma inversa, a curva senoidal  $\rho = x \cos(\theta) + y \sin(\theta)$  passa pelo ponto  $(\rho_o, \theta_o)$  no plano- $\rho\theta$  se  $(x, y)$  pertence a linha  $(\rho_o, \theta_o)$  no plano- $xy$ .

De modo a ilustrar o processo da THP com funções senoidais, repete-se o exemplo da seção anterior, como ilustra a Figura 4.4

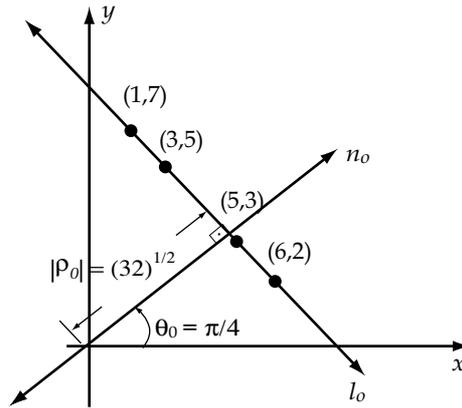


Figura 4.4: Exemplo de aplicação da THP em um conjunto de quatro pontos que definem uma reta com  $\rho = \sqrt{32} \approx 5,657$  e  $\theta = \pi/4 \approx 0,7854$ .

A THP mapeia os pontos indicados em funções senoidais:

$$h : (1, 7) \rightarrow \rho = 1 \cos(\theta) + 7 \sin(\theta)$$

$$h : (3, 5) \rightarrow \rho = 3 \cos(\theta) + 5 \sin(\theta)$$

$$h : (5, 3) \rightarrow \rho = 5 \cos(\theta) + 3 \sin(\theta)$$

$$h : (6, 2) \rightarrow \rho = 6 \cos(\theta) + 2 \sin(\theta)$$

Os gráficos destas funções senoidais estão representados na Figura 4.5

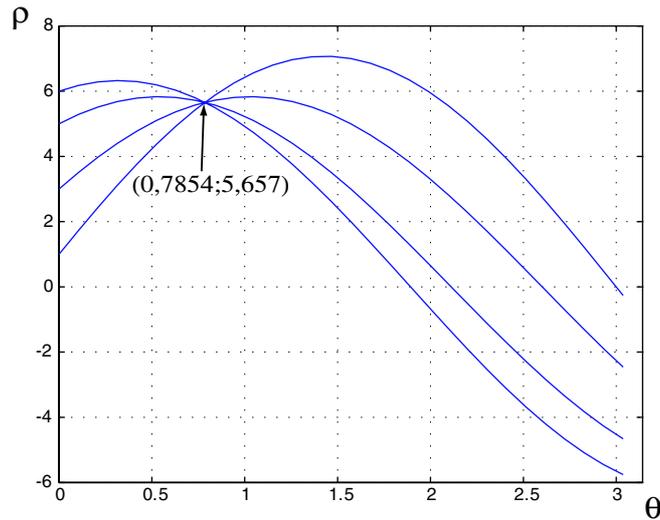


Figura 4.5: Exemplo do espaço  $\rho, \theta$ . Os quatro pontos do plano- $xy$  tornaram-se funções senoidais no plano- $\rho\theta$ . O ponto de interseção indica o coeficiente angular e linear da reta no plano- $xy$ .

Cada ponto  $(x, y)$  no domínio do espaço de dados é mapeado para funções senoidais no espaço de parâmetros. Se um conjunto de pontos  $(x_i, y_i)$  do espaço de dados pertencer a uma reta no plano- $xy$ , no espaço de parâmetros as funções senoidais se interceptarão no ponto  $(\rho_o, \theta_o)$ . De forma semelhante, uma função senoidal  $\rho = x \cos(\theta) + y \sin(\theta)$  interceptará  $(\rho_o, \theta_o)$  somente se a localização do ponto  $(x, y)$  for ao longo da reta  $(\rho_o, \theta_o)$  no plano- $xy$ . Portanto é possível contar o número de pontos que estão em  $(\rho_o, \theta_o)$  no plano- $xy$ , através da contagem do número de funções senoidais no plano- $\rho\theta$  que se interceptam no ponto  $(\rho_o, \theta_o)$ . Esta observação é a base da Transformada de Hough e sua analogia também é válida para o plano- $mc$ .

### 4.1.3 Discretização da Transformada de Hough Padrão

De modo a contar o número de interseções das retas ou das funções senoidais tem-se que discretizar o espaço de parâmetros  $mc$  ou  $\rho\theta$ . Para tal, devemos estipular a faixa de variação dos parâmetros e o número de divisões, ou elementos. Essa representação discreta do espaço de parâmetros é chamada de matriz de acumulação (a). O ponto de máximo deste acumulador indica os parâmetros que definem a reta no plano- $xy$ .

A determinação da granularidade do acumulador do espaço de parâmetros requer um compromisso entre precisão e custo computacional. Quanto mais fina a granularidade menor será o desvio padrão estabelecido por um elemento, entretanto maior será o custo computacional na formação da matriz de acumulação  $\mathbf{a}$ . Outra característica do aumento da granularidade é a ocorrência de diversos pontos de máximo muito próximo, representando erroneamente a ocorrência de mais de uma reta no espaço de dados.

Caso a faixa de variação de parâmetros seja conhecida teremos:

$$-M \leq m \leq M, \quad -C \leq c \leq C$$

onde  $M$  e  $C$  representam os limites de variação do espaço de parâmetros.

Com passo de variação, ou número de elementos, especificado pelas variáveis  $n_m$  e  $n_c$ . Por exemplo, se desejamos uma variação de 0,1 em  $m$  e 1 em  $c$ , obtemos  $n_m$  e  $n_c$  através das equações 4.4 e 4.5

$$n_m = \left\lceil \frac{\Delta_m}{p_m} \right\rceil \quad (4.4)$$

$$n_c = \left\lceil \frac{\Delta_c}{p_c} \right\rceil \quad (4.5)$$

onde  $\Delta_m = M - (-M)$ ,  $\Delta_c = C - (-C)$ ,  $p_m = 0,1$  e  $p_c = 1$ .

Para o caso senoidal, supondo que desejamos uma variação de 0,1 em  $\rho$  e 0,3 em  $\theta$ , a determinação da matriz  $\mathbf{a}$  será:

$$-R \leq \rho \leq R \quad 0 \leq \theta \leq \pi$$

O número de elementos é especificado por:

$$n_\rho = \left\lceil \frac{\Delta_\rho}{p_\rho} \right\rceil \quad (4.6)$$

$$n_\theta = \left\lceil \frac{\Delta_\theta}{p_\theta} \right\rceil \quad (4.7)$$

onde  $\Delta_\rho = R - (-R)$ ,  $\Delta_\theta = \pi$ ,  $p_\rho = 0,1$  e  $p_\theta = 0,3$ .

A matriz de acumulação terá dimensão  $n_m \times n_c$  ou  $n_\rho \times n_\theta$ .

Inicialmente, todas as células do acumulador são zeradas. O valor do acumulador  $\mathbf{a}(i, j)$  é incrementado de 1 para cada coordenada  $(x, y)$  de entrada que satisfizer a inequação:

$$|\rho_i - (x \cos(\theta_j) + y \sin(\theta_j))| < \varepsilon \quad (4.8)$$

$$|c_i + m_j x - y| < \varepsilon \quad (4.9)$$

onde  $\varepsilon$  é um fator de erro utilizado para compensar os erros de quantização e digitalização. Isto significa que, se o ponto  $(c_i, m_j)$  ou  $(\rho_i, \theta_j)$  estiver na reta  $c = -mx + y$  ou na curva  $\rho = x \cos(\theta) + y \sin(\theta)$  ( dentro de uma margem de erro ), a célula do acumulador na posição  $(i, j)$  é incrementada. Esta margem de erro normalmente é igual à variação de um elemento.

Quando o processo de incremento da matriz de acumulação termina, cada valor de  $\mathbf{a}(i, j)$  será igual ao número de funções que interceptaram o ponto  $(c_i, m_j)$  ou  $(\rho_i, \theta_j)$  no espaço de parâmetros. Conforme vimos anteriormente, este é o número de pontos no espaço de dados que estão na linha definida pelos coeficientes angular  $m$  e linear  $c$ .

#### 4.1.4 Presença de Ruído no Espaço de Dados

Nos exemplos apresentados até aqui todos os pontos do espaço de dados eram colineares, formando um único ponto de interseção no espaço de parâmetros de Hough. Na prática, isto não acontece. Mesmo que os pontos pertençam a uma única reta, haverá sempre distorção ou ruído.

Basicamente dois tipos de ruído podem ocorrer no espaço de dados. O primeiro é o ruído de fundo e o segundo é o de distorção da reta ou do objeto em estudo.

O ruído de distorção vai ocasionar, no acumulador, a existência de pequenas torres espalhadas. Neste caso não há um único ponto de interseção, mas um aglomerado de interseções. Existem diversas maneiras de determinar o ponto de interseção no espaço de parâmetros. Uma forma é calcular o ponto de médio ou o centróide da nuvem de pontos, como na equação:

$$\begin{aligned} \theta_c &= \frac{\sum_i m a_i \theta_i}{\sum_i m a_i} \\ \rho_c &= \frac{\sum_i m a_i \rho_i}{\sum_i m a_i} \end{aligned} \quad (4.10)$$

O ruído de fundo no espaço de dados produz no acumulador diversas torres espalhadas. Este tipo de ruído, no entanto, não modifica a torre composta pelos pontos colineares da reta. Desta forma, a localização da torre de maior pico do histograma bidimensional continuará indicando os coeficientes da reta. Portanto, o inconveniente da presença do ruído no espaço de dados é o aumento do custo computacional para processar dados que não representam a estrutura procurada.

#### 4.1.5 Considerações

A THP consiste em uma ferramenta simples e eficaz de detecção de objetos em imagens, que possam ser descritos por equações. Esta seção apresentou exemplos de detecção de retas, no entanto, poderia ser qualquer equação polinomial. O espaço de parâmetros varia conforme o grau da equação.

Foi verificado que a THP utiliza um espaço de parâmetros discreto e que a determinação dos limites deste espaço tem que ser estabelecida a priori. Esta determinação implica no conhecimento antecipado dos objetos que deverão ser detectados no espaço de dados. Como na maioria das aplicações reais não será possível o conhecimento a priori dos objetos, matrizes de grandes dimensões devem ser criadas no espaço de parâmetros. Desta forma, há necessidade de muita memória de armazenamento e um grande esforço computacional para localizar aglomerações e seus respectivos centróides.

Esses fatores limitam a utilização da THP. Nas seções seguintes serão apresentadas algumas modificações na THP de modo a tornar a Transformada de Hough uma ferramenta mais eficiente e rápida de detecção de equações em imagens.

## 4.2 Transformada de Hough Local (THL)

A transformada local [63] surge como alternativa à forma padrão e fornece alguns recursos para resolver algumas deficiências da THP.

### 4.2.1 Características

Na THP a conversão do espaço de dados para o de parâmetros ocorria através da operação de retroprojeção. Esta operação consistia em, para cada ponto do

espaço de dados, construir uma reta ou senóide no espaço projetado, de forma que a interseção de muitas retas ou senóides em um dado ponto do espaço de parâmetros indicava a presença de uma reta no espaço de dados. Em outras palavras, era feito um “mapeamento divergente” do espaço de dados para o espaço de parâmetros. A divergência ocorre porque um único ponto no espaço de dados gerava vários pontos no espaço de parâmetros.

Na intenção de reduzir de maneira significativa os gastos computacionais com memória e tempo de processamento que a THP requeria, surgiu a idéia da THL. A idéia principal da THL consiste em encontrar a melhor reta que pode ter gerado um determinado grupo de pontos no espaço de dados. Desta forma, cada par de pontos no espaço de dados é mapeado em um único ponto no espaço de parâmetros. Portanto, a THL realiza um “mapeamento convergente”, um vez que mapeia um par de pontos do espaço de dados em um único ponto no espaço de parâmetros[63].

A construção do acumulador para o caso da THL dispensa a necessidade de se fazer uma varredura no espaço de parâmetros. Para cada par de pontos do espaço de dados, apenas uma célula do acumulador é incrementada. Se dois pares distintos de pontos pertencerem à mesma reta no espaço de dados, eles vão ocasionar o incremento da mesma célula no acumulador, já que o ponto mapeado no espaço de dados será o mesmo. Desta forma, os picos no acumulador, tal como na THP, representam grandes probabilidades da existência de uma reta no espaço de dados com os parâmetros dados pelas coordenadas do pico.

O ponto no espaço de parâmetros é dado pela solução do sistema de duas equações e duas incógnitas, equação 4.11. A Figura 4.6 apresenta uma ilustração da escolha de pares de pontos da THL. A Figura 4.7, por sua vez, apresenta o fluxograma do algoritmo da THL.

$$y_1 = mx_1 + c \quad (4.11)$$

$$y_2 = mx_2 + c \quad (4.12)$$

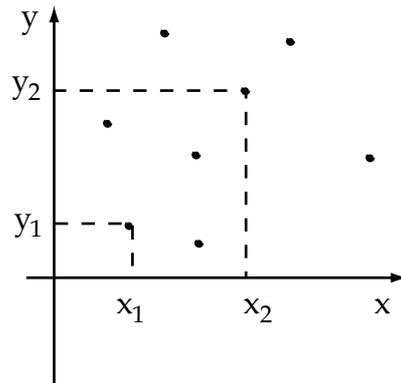


Figura 4.6: Seleção de um par de pontos para o cálculo da THL.

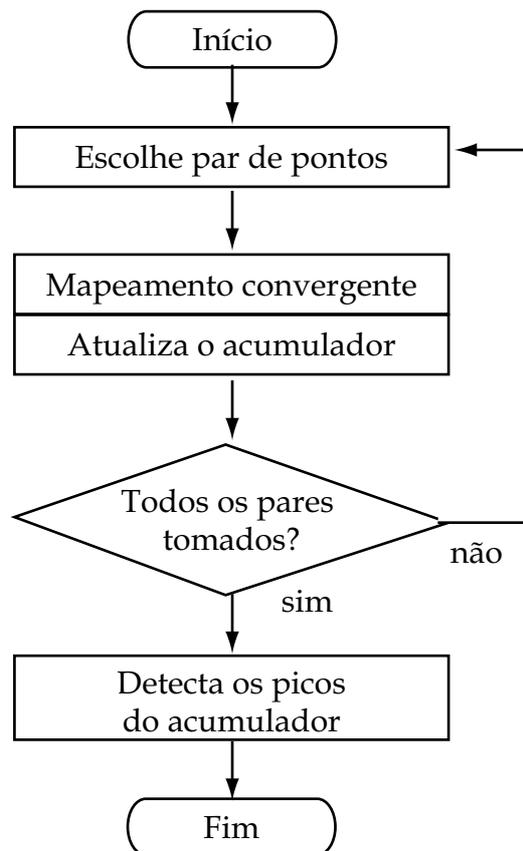


Figura 4.7: Fluxograma para a THL.

## 4.2.2 Considerações

A THL possui algumas vantagens e restrições. Uma restrição consiste no fato da THL não priorizar o mapeamento de estruturas de dados que se encontrem sobre todo o espaço de parâmetros, uma vez que seu processamento é feito com pares de pontos, buscando uma informação local apenas. O processo de seleção de vários destes pares, de maneira a percorrer todo o espaço de dados, ameniza esta restrição, e estruturas globais podem ser bem reconstruídas[71].

A THL traz muitas vantagens para aplicações em que se quer detectar retas ou curvas em imagens. Grande parte dos benefícios de sua utilização se aplicam a todos os casos em que se faz uso do processamento da transformada de Hough.

Os principais são:

- A THL permite maior rapidez no processamento que a THP. Isto se deve basicamente a dois fatores: primeiramente, é mais fácil fazer o mapeamento de pares de pontos em um outro ponto, do que determinar todas as curvas que passam por um dado ponto (mapeamento convergente  $\times$  Mapeamento divergente).
- Menor necessidade de memória. Na THL, a necessidade de armazenamento no acumulador é menor, mesmo se o acumulador tiver mesmo tamanho do construído para a THP.
- Menos “ruído” no espaço de parâmetros. A técnica empregada na THP faz com que um ponto no espaço de dados cause o incremento de várias células no espaço de parâmetros, sendo que apenas uma delas representará o conjunto de parâmetros que se quer determinar, uma vez que um dado ponto pertence a uma reta somente. O ruído causado pela THP, além de poder eventualmente dificultar a detecção de picos no acumulador, representa um enorme e desnecessário custo computacional, isto é, grande parte do tempo gasto no processamento da THP deve-se ao cálculo de curvas que nunca poderiam representar uma curva verdadeira no espaço de dados. Na THL isto não acontece, porque somente são processadas as retas que tem possibilidade de representar alguma estrutura real no espaço de dados[72].

## 4.3 Transformada de Hough Adaptativa (THA)

Uma outra variação para a transformada de Hough é a transformada de Hough Adaptativa (THA). Ela foi introduzida por J. Illingworth e J. Kittler em artigo intitulado “*The Adaptive Hough Transform*”, de 1987[73].

### 4.3.1 Características

A idéia principal da THA consiste em realizar repetidamente o processamento da THP ou THL, para um determinado espaço de dados, sendo que a cada iteração é feito um *zoom* no acumulador, na região vizinha ao pico detectado. O princípio da THA baseia-se no fato de que, no início do processamento, não se sabe em que região do espaço de parâmetros vai se localizar o pico máximo representando a reta procurada no espaço de dados. Desta forma, constrói-se inicialmente um acumulador com poucas células, e, portanto de resolução pouco fina, fazendo com que o processamento seja bastante rápido e logo se ache um pico. Após a realização do processamento da THP ou THL para este acumulador inicial e da determinação da célula de maior altura (pico), é feita a reconstrução do acumulador de maneira que em seu centro fique a célula “vencedora” da iteração anterior. Desta forma, a iteração seguinte consistirá do processamento da transformada sobre um novo acumulador, de maior precisão, em que será detectada uma nova célula vencedora, a ser utilizada na construção do próximo acumulador.

A repetição destes passos faz com que o pico inicialmente detectado seja identificado com cada vez mais precisão. O critério de parada para o algoritmo consiste em definir uma precisão desejada na determinação dos parâmetros e verificar, ao final de cada iteração, se tal precisão já foi atingida ou se um maior refinamento é ainda necessário. A Figura 4.8 consiste de um fluxograma, onde o algoritmo da THA é formalizado.

### 4.3.2 Considerações

As principais vantagens da THA referem-se aos ganhos computacionais obtidos. De fato, apesar de serem necessárias várias iterações da transformada para o processamento do espaço de dados, a soma do tempo destas iterações é em geral

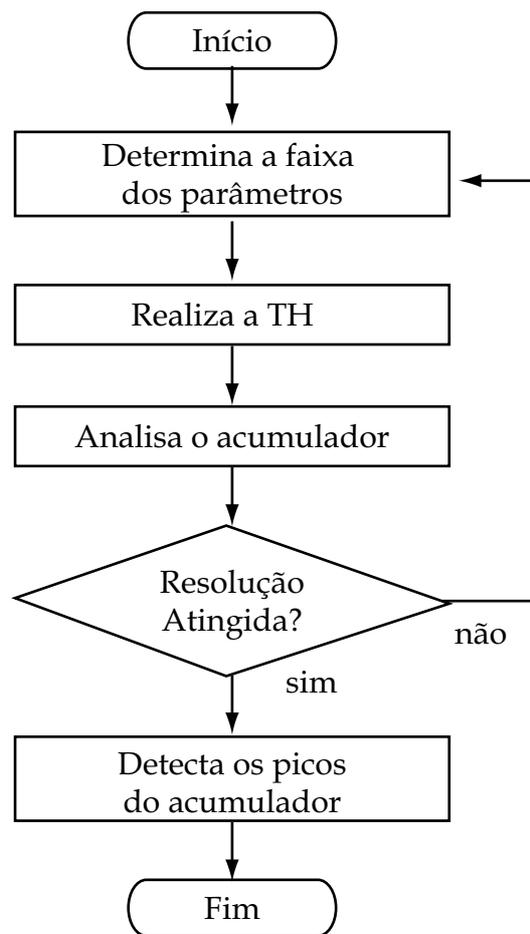


Figura 4.8: Fluxograma para o algoritmo da THA.

menor do que o tempo necessário para computar, de uma só vez, o acumulador fixo utilizado na THP e na THL, notadamente quando se deseja elevada acurácia na determinação dos parâmetros da reta. Além disso, a necessidade de armazenamento na THA é bastante reduzida em relação às outras transformadas (principalmente a THP), uma vez que o tamanho dos acumuladores utilizados em cada iteração não precisa ser muito grande (devido ao *zoom*) e não há a necessidade de que acumuladores sejam armazenados de uma iteração para outra.

Considerando-se o caso da detecção de retas, faz-se aqui uma comparação entre as eficiências relativas da THP e da THA. Se cada dimensão do acumulador fixo usado na transformada padrão, tem  $\alpha$  células, contra  $\beta$  células do acumulador usado na THA, a razão entre os tamanhos dos dois acumuladores é:

$$R_s = \left(\frac{\alpha}{\beta}\right)^2 \quad (4.13)$$

Esta razão, dependendo dos valores de  $\alpha$  e  $\beta$ , pode ser bastante grande. Porém, para atingir a mesma resolução da THP, são necessárias várias iterações da THA. A resolução na THA aumenta na medida em que se diminui a faixa dinâmica dos parâmetros para a construção do novo acumulador de cada iteração (processo de *zoom*). Se cada um dos parâmetros na THA tem sua faixa reduzida de um fator  $\frac{1}{\gamma}$  a cada iteração, o número de iterações  $l$  necessárias para que a THA alcance a mesma resolução da THP vem de:

$$\left(\frac{1}{\gamma}\right)^l = \frac{1}{\alpha} \quad (4.14)$$

Logo:

$$l = \frac{\log(\alpha)}{\log(\gamma)} \quad (4.15)$$

Considerando agora que o número  $N$  de pontos do espaço de dados é bastante grande, a quase totalidade do tempo gasto no processamento da transformada é devido à operação de retroprojeção e à construção do acumulador. No caso da THP, a varredura utilizada na construção do acumulador requer que se calculem  $\alpha$  valores para um dos parâmetros, enquanto se varia o outro, dentro de sua faixa dinâmica de  $\alpha$  possíveis valores. Como o número de verificações de interseções entre as curvas projetadas e as células do acumulador é de  $2(\beta+1)$  para cada iteração da THA, tem-

se que uma única iteração sua é da ordem de  $\frac{\alpha}{2(\beta+1)}$  mais rápida que a computação da THP. Desta forma, a eficiência relativa entre a THA e a THP é dada por:

$$R_c = O\left(\frac{\alpha}{2(\beta+1)} \frac{\log(\gamma)}{\log(\alpha)}\right) \quad (4.16)$$

Para os casos em que há somente uma reta no espaço de dados, esta variação da transformada é a mais eficiente. No entanto, ela começa a não ser muito vantajosa a partir do momento em que há no espaço de dados um número grande de retas. Isto acontece porque, ao final de cada iteração, tem que ser aberta uma “janela” para cada pico detectado e se fazer o *zoom* para cada um dos acumuladores gerados. Para a presença de  $q$  retas, a eficiência relativa  $R_c$  cai para:

$$R_c = O\left(\frac{\alpha}{2q(\beta+1)} \frac{\log(\gamma)}{\log(\alpha)}\right) \quad (4.17)$$

e a razão entre os tamanhos dos acumuladores fica igual a  $\frac{\alpha^2}{q\beta^2}$ . Como na THA a resolução inicial do acumulador é bastante baixa, pode não ser possível distinguir entre retas com parâmetros próximos. Desta forma, um único pico inicial pode representar mais de uma curva. Nas iterações seguintes, com os acumuladores mais precisos, possivelmente ocorrerá mais de um pico. Neste caso, não há como determinar qual dos picos é o correto ou desejado, sendo necessário abrir mais de uma janela de recursão. Portanto, a determinação do ponto de parada da THA não é trivial para casos de imagens com mais de uma reta ou muito ruidosas[73].

## 4.4 Transformada de Hough Morfológica (THM)

O surgimento da THM foi baseado na filtragem morfológica de imagens. Em 1997 Schmidt[74] utilizou as operações morfológicas de erosão e dilatação pela primeira vez no espaço de parâmetros de Hough objetivando reduzir o ruído e aumentar a eficiência na detecção de retas em imagens. Neste método o elemento estruturante (ou máscara morfológica) da THM, a Figura 4.9 ilustra alguns exemplos de máscara, percorre todo o espaço de parâmetros de Hough realizando a operação de abertura (erosão seguida pela dilatação). Esta operação morfológica elimina os ruídos espúrios do espaço de parâmetros, aumentando a eficiência de localização de retas no espaço de dados de Hough.

Em 2004 Sim[75] utilizou a THM numa implementação da transformada de Hough em sistemas híbridos. A arquitetura de processamento paralela do sistema híbrido em questão utilizava processadores de instruções simples e múltiplos dados (SIMD) e instruções e dados múltiplos (MIMD). O objetivo consistia em verificar a eficiência e o tempo de processamento da THM neste sistema híbrido. Foi observado que pequenas retas foram detectadas, sem alterar a eficiência de detecção das retas maiores. Isto ocorreu pois a imagem do espaço de parâmetros foi dividida em pequenas imagens e processadas de forma paralela. Foi observado também uma redução de 5,5 vezes no tempo de processamento.

Nessa Tese a transformada de Hough morfológica é utilizada como uma variação da transformada de Hough adaptativa. Isto se deve ao fato da máscara morfológica ser utilizada somente no elemento de maior contagem do espaço de parâmetros com o objetivo de decidir quais elementos serão considerados na recursão seguinte da transformada. Neste caso, a morfologia matemática não é utilizada no espaço como um todo para que haja redução de ruído, como fora em [74] e [75]. Ela é utilizada pontualmente, com centro do elemento estruturante no elemento de maior contagem do espaço de parâmetros.

As próximas seções descrevem a THM, onde serão apresentados: a descrição de seus mecanismos e suas vantagens; o algoritmo, esquematizando-o por meio de fluxogramas simplificados e a análise quantitativa de alguns de seus aspectos.

#### **4.4.1 Características**

A implementação do método THM neste trabalho, como todas as demais formas da transformada de Hough, tem como objetivo a localização de vértices primários a partir de imagens contendo traços. Basicamente, o trabalho a ser feito consiste em determinar a existência de um vértice em uma dada imagem. Em um primeiro momento, a THM pode ser vista como uma modificação do método contido na THA.

##### **4.4.1.1 Mecanismos e Vantagens**

O mecanismo da THM consiste em utilizar a máscara morfológica (ou elemento estruturante) no elemento de maior contagem do espaço de parâmetros. O

centro da máscara é posicionado no elemento de maior contagem. Todos os elementos da máscara de valor igual a 1 determinam os elementos do espaço de parâmetros a serem considerados. Os elementos considerados como relevantes serão utilizados na recursão seguinte do espaço de parâmetros. Isto representa um ajuste do espaço de parâmetros, onde as características morfológicas do espaço são consideradas relevantes[76, 77].

Na THA a recursão ocorre somente no elemento de maior contagem, sem ser levado em consideração seus vizinhos. A THM, no entanto, objetiva utilizar esta informação da vizinhança através da aplicação de uma máscara ( $\mathbf{M}$ ) morfológica no elemento de maior contagem.

A vantagem deste método com relação à THA e a THL consiste em utilizar a característica da morfologia do espaço de parâmetros de Hough.

#### 4.4.1.2 As Diferentes Formas de $\mathbf{M}$

A máscara  $\mathbf{M}$  deve ser ajustada de acordo com a característica do espaço de parâmetros. No caso específico do sistema de localização de vértices de colisão, a correlação do ruído no espaço de dados provoca no espaço de parâmetros um conjunto de elementos próximos com contagens elevadas. Esses elementos normalmente são vizinhos no parâmetro do coeficiente linear e possuem o mesmo coeficiente angular.

A escolha do elemento estruturante foi realizada levando em consideração essa característica dos vizinhos do espaço de parâmetros. Diversas máscaras com diferentes tamanhos e formas foram testadas no conjunto de dados. Os resultados obtidos na eficiência do sistema foram comparados e as máscaras com as melhores eficiências foram selecionadas. A Figura 4.9 apresenta algumas das máscaras utilizadas no sistema de localização de vértices de colisão. As máscaras realçam os vizinhos horizontais ao elemento central, uma vez que as máscaras possuem mais pixels na horizontal.

A Figura 4.10 representa o fluxograma da THM. Assim como na THA, a formação do espaço de parâmetros pode ser convergente ou divergente. Em seguida é realizada a análise do acumulador, que consiste em encontrar o elemento de maior contagem do espaço de parâmetros. A máscara morfológica é aplicada neste elemento, determinando quais são as novas faixas de parâmetros e também selecio-

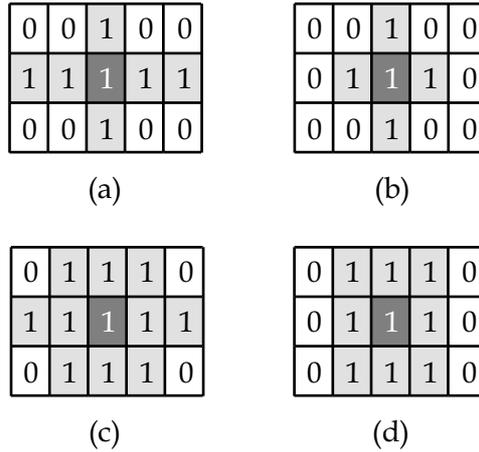


Figura 4.9: Ilustração das máscaras utilizadas na THM.

nando os pontos do espaço de dados que serão utilizados na recursão seguinte. Uma vez atingida a condição de parada é calculado o valor do vértice  $z_{vtx}$ [78].

#### 4.4.2 Considerações

As principais vantagens da THM consistem em aliar as vantagens da THA com a informação característica do espaço de parâmetros. Dessa forma, a THM melhora a precisão no espaço de parâmetros, permanecendo com tempo de execução da THA.

A diferença entre a THA e a THM está na máscara ou elemento estruturante. A utilização da máscara define os pontos no espaço de dados que serão utilizados para formar o espaço de parâmetros da recursão seguinte. A máscara também define o limite do espaço de parâmetros. A máscara pode ser empregada em uma ou mais recursões. A abordagem da máscara também pode ser empregada na THL. No entanto, um estudo da forma e tamanho da máscara deve ser realizado.

O aspecto negativo da THM consiste na determinação de uma máscara que seja mais adequada aos dados de entrada. Para que esta transformada seja utilizada é necessário que o espaço de parâmetros tenha um comportamento semelhante para todos os eventos de entrada.

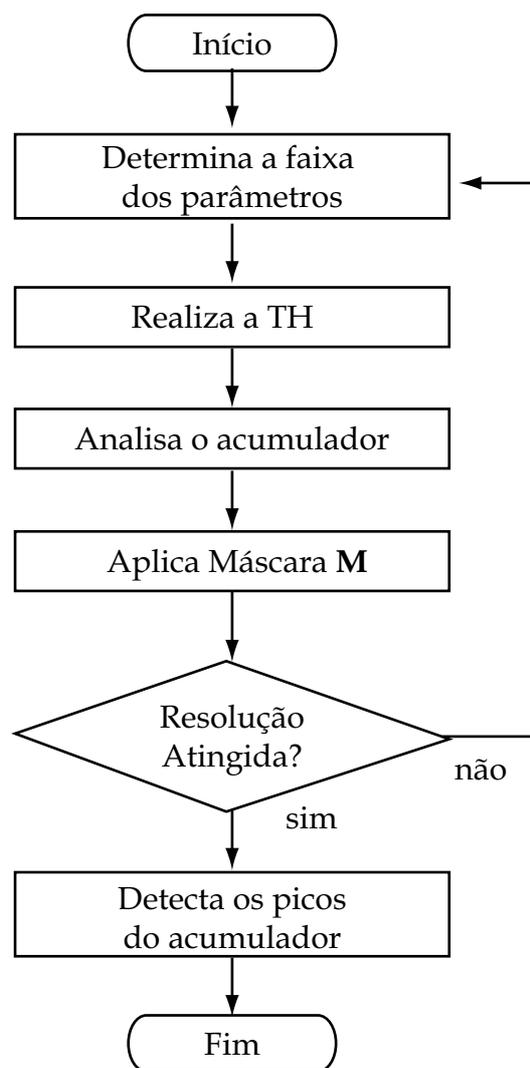


Figura 4.10: Fluxograma para o algoritmo da THM.

# Capítulo 5

## O Conjunto de Dados

Este capítulo objetiva apresentar os dados de entrada utilizados nessa tese. Conforme mencionado no capítulo 3, sistemas de reconstrução de vértices utilizam as informações dos detectores de traço para a localização da origem de colisões. No caso específico desse trabalho a entrada consiste na imagem volumétrica dos detectores PIXEL e SCT do ATLAS.

Inicialmente este capítulo apresenta a forma como os dados de entrada são formados. Em seguida, aborda a forma de validação e armazenamento dos conjuntos. Por fim, apresenta como são produzidas as imagens volumétricas e a forma como estas serão utilizadas na tese.

### 5.1 A Geração dos Dados

Durante a fase de preparação do LHC, todos os experimentos apresentam grande necessidade de dados simulados, tanto para o projeto quanto para a otimização dos detectores e os algoritmos de análise. O experimento ATLAS conta com uma equipe de desenvolvimento, validação e armazenamento de simulações dos dados de colisão e das respostas dos detectores.

A simulação Monte Carlo é utilizada no processo inicial da construção dos dados, como ilustra a Figura 5.1. A construção dos arquivos é dada da seguinte forma[79]:

- As partículas provenientes das colisões, chamadas de estado final de colisão ou simplesmente estado final, são geralmente obtidas utilizando os programas

baseados nas teorias e na fenomenologia da física (chamados geradores);

- As partículas do estado final gerado são transportadas através do detector virtual de acordo com as leis conhecidas da física, que governam a passagem das partículas através da matéria;
- As interações resultantes com os elementos sensíveis do detector são digitalizadas de forma similar àquelas produzidas pelo detector real;
- O evento é reconstruído;
- A informação Monte Carlo que gerou o evento, também chamada de *truth* ou *true*, é conservada para a comparação com a informação reconstruída.

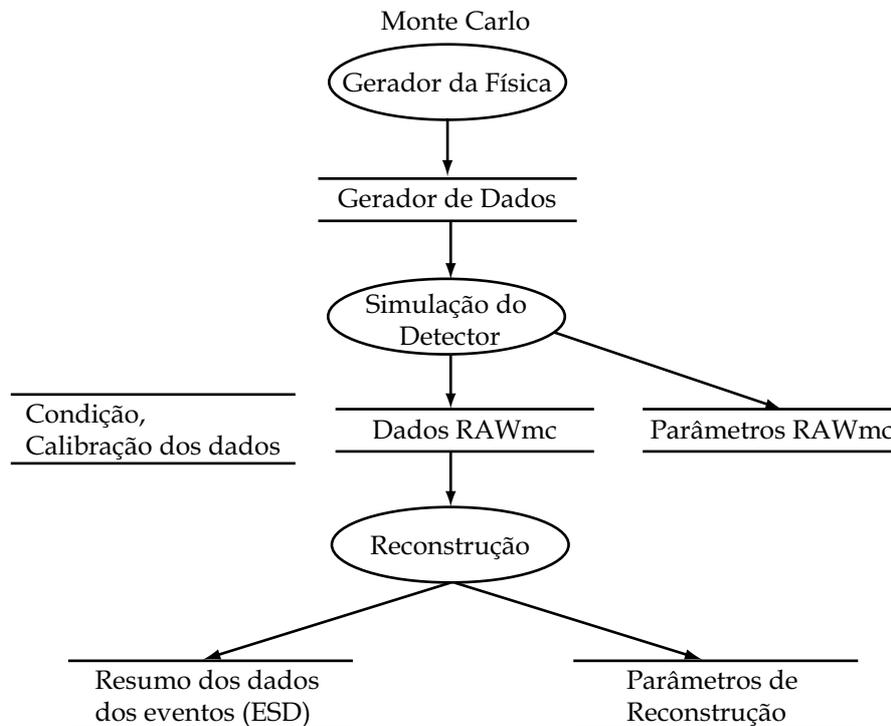


Figura 5.1: Esquema do processo de simulação de dados no ATLAS.

A geração de todas as amostras de eventos feitas no CERN utilizam o Pythia 6.2031[80] e são executadas no ATHENA.

Na luminosidade projetada ( $10^{34} \text{cm}^{-2} \text{s}^{-1}$ ), a média de eventos no ATLAS será igual a 23 por colisão. Este número varia de acordo com a distribuição de Poisson[81]. O empilhamento de eventos ocorre devido ao grande número de pontos e varia de acordo com o tempo de memória de uma colisão. Nos detectores de silício,

o tempo de memória é de 1 colisão; no entanto, nos calorímetros e nos detectores de múons este tempo sobe para 30 colisões.

Para gerar sinais de eventos com a presença de empilhamento, eventos de colisões são adicionados. O número de colisões “vistas” por um subdetector é especificada individualmente. Nos detectores de traço, uma colisão é vista pelos detectores PIXEL e SCT, enquanto que o detector TRT é sensível a duas colisões[79].

De modo a gerar sinais de alta luminosidade numa colisão são adicionadas amostras de empilhamento em cada um dos detectores. O tempo de interação dos eventos de empilhamento é aleatório para cada novo evento. Depois de cada colisão aproximadamente 10% do evento é armazenado para ser adicionado a outro evento como empilhamento.

Qualquer colisão registrada nos detectores do ATLAS contém superposição de partículas. Estas partículas são oriundas de outra colisão *pp*. Na média, por cada colisão observada nos detectores do ATLAS. O número total de partículas observadas por evento depende do sinal de tempo de coleta, que varia de poucos *ns* para os detectores de silício até aproximadamente  $700ns$  para os detectores de múons. No calorímetro LAr o sinal é medido logo após a filtragem, de modo que ele é afetado somente pela colisão anterior, medidas nos detectores TRT e de múons (MDT) continuam por um tempo tal que eles são sensíveis a colisões posteriores.

O processo completo de simulação de empilhamento corresponde à sobreposição de colisões no tempo e nas simulações da física. Como toda colisão é normalmente simulada para  $100ns$  de tempo de propagação, existe um componente adicional faltando nesse esquema. Este componente vem dos nêutrons que podem passar pela caverna do ATLAS por alguns segundos até que se termalizem. Este componente, *i.e.*, pontos adicionais criados por partículas de vida longa, forma o ruído de fundo da caverna (*cavern background*).

O ruído de fundo da caverna é simulado separadamente, como um componente que é adicionado sobre todo evento de elétrons individuais (*single-e*).

## 5.2 Caracterização do Vértice Primário de Colisão

A trajetória de uma partícula carregada dentro de um campo magnético uniforme é uma hélice. Em coordenadas cilíndricas, com o eixo  $z$  paralelo ao campo magnético, a hélice passa pelo vértice com coordenadas  $(0, 0, z_v)$  e pode ser descrita pela equação 5.1.

$$\rho = Gp_t \sin(\phi - \phi_0) \cong Gp_T(\phi - \phi_0) = G'(\phi - \phi_0) \quad (5.1)$$

$$z - z_v = Gp_z(\phi - \phi_0) \quad (5.2)$$

onde  $p_T$  e  $p_z$  são os momentos transverso e longitudinal,  $G$  é uma constante conhecida que depende do campo magnético e  $G' = Gp_T$ . A partir destas equações uma terceira pode ser obtida, equação 5.3,

$$z - z_v \cong \frac{p_z}{p_T} \rho = C\rho \quad (5.3)$$

que é linear em uma boa aproximação. O vértice primário de colisão quando observado na projeção  $(\rho, z)$  pode ser localizado através da equação de reta 5.4.

$$z = C\rho + z_v \quad (5.4)$$

## 5.3 Ilustrando os Eventos

A imagem volumétrica de entrada contém informação da granularidade completa dos detectores de traço. Duas tecnologias diferentes são utilizadas, que são: a de pixel e a de silício.

O detector PIXEL possui três barris cilíndricos concêntricos e oito discos de detecção, quatro de cada lado, como ilustra a Figura 5.2. Os barris são compostos por módulos de dimensão idêntica. Cada módulo tem  $64 \text{ mm} \times 22,4 \text{ mm}$  e possui 61440 *pixels*. Os módulos são sobrepostos, como ilustra a Figura 5.3, de modo a realizar uma cobertura hermética.

O sistema SCT é projetado para realizar medidas de precisão utilizando tiras de silício. Ele é composto por quatro barris concêntricos e seis discos de detecção.

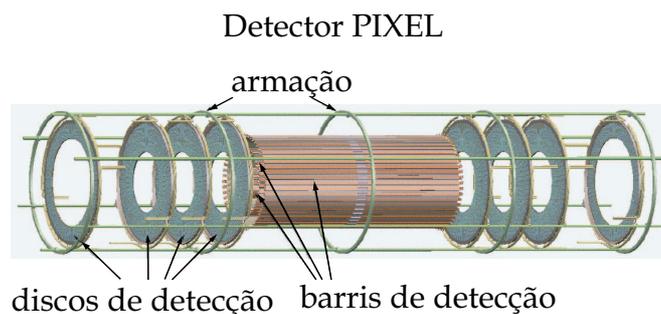


Figura 5.2: Esquema tridimensional do detector PIXEL.

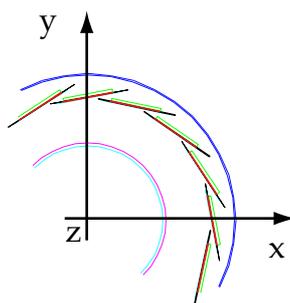


Figura 5.3: Primeiro barril do detector PIXEL, com 4cm de raio. Os elementos sobrepostos são os módulos.

Assim como no detector PIXEL, o SCT é formado por módulos. Cada módulo possui quatro elementos sensíveis, dois de cada lado, que são chamados de tiras de silício. A resolução espacial do SCT é de  $16 \mu\text{m}$  na direção  $\rho\phi$  e  $580 \mu\text{m}$  na direção  $z$ . Trajetórias podem ser distinguidas se estiverem a uma distância superior a  $\approx 200\mu\text{m}$ .

De modo a utilizar a granularidade mais fina dos detectores de traço para a localização *online* da origem da colisão, esta Tese utiliza o conceito de regiões de seleção e de pontos tridimensionais. Isto significa que a imagem volumétrica é composta somente pela informação tridimensional dos elementos sensíveis sensibilizados quando da passagem de uma partícula. Para utilizar os dados de entrada nesta configuração foi desenvolvido um programa de seleção de regiões, abordado no Capítulo 6. A Figura 5.4 ilustra três vistas diferentes do evento (a)  $\phi \times \rho$ , (b)  $\rho \times z$  e (c)  $\phi \times z$ . Desta forma, é utilizada a granularidade completa dos detectores e a imagem volumétrica é composta por uma lista de pontos tridimensionais.

As técnicas de localização do vértice primário de colisão apresentadas no Capítulo 8 utilizam como dados de entrada dos métodos esta lista de pontos tridimensionais. Este procedimento reduz significativamente o tempo de processamento

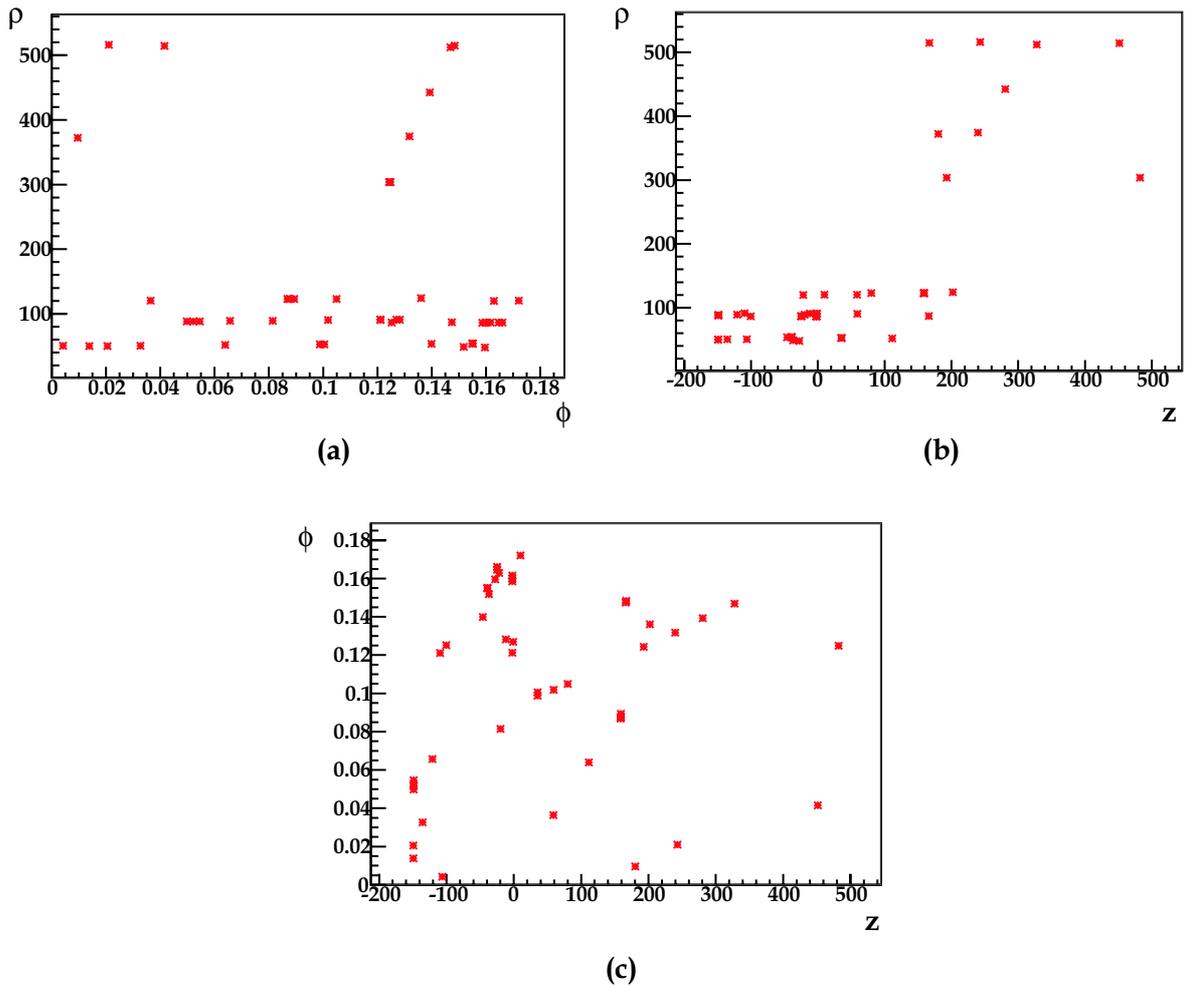


Figura 5.4: Três vistas do evento de elétron individual com baixa luminosidade nos detectores de traço PIXEL e SCT. (a)  $\phi \times \rho$ . (b)  $\rho \times z$  e (c)  $\phi \times z$ .

	Baixa luminosidade $2 \times 10^{33} \text{cm}^{-2} \text{s}^{-1}$	Alta luminosidade $10^{34} \text{cm}^{-2} \text{s}^{-1}$	$2 \times 10^{33} \text{cm}^{-2} \text{s}^{-1}$
Radiação <i>Bremsstrahlung</i>	não	sim	sim
Número de eventos	10000	10000	5000
Energia transversa	20 GeV	30 GeV	25 GeV
Tipo evento	elétrons		
$ \eta $	< 2,5		

Tabela 5.1: Características dos dados utilizados.

dos métodos de reconstrução, uma vez que não são utilizados os vóxels iguais a zero da imagem volumétrica dos detectores.

## 5.4 Conjunto de Dados Utilizado

Nessa tese, foi utilizado o conjunto de dados de elétrons (*single-e*). Os dados correspondem a simulações do DC1 com empilhamento de eventos (*pile-up*) com alta e baixa luminosidade. A Tabela 5.4 apresenta as características relevantes dos eventos testados.

O empilhamento ocorre com menor probabilidade em baixa luminosidade do que em alta luminosidade, possuindo uma média de 4,6 regiões de interesse por colisão enquanto que, na alta luminosidade, a média é de 23 regiões de interesse por colisão. A Figura 5.5(a) ilustra um evento de empilhamento a baixa luminosidade, e 5.5(b) um evento de empilhamento a alta luminosidade vistos pelos detectores de traço PIXEL e SCT.

Em média, elétrons com empilhamento e baixa luminosidade, possuem 43 ( $\pm 25, 6$ ) pontos. No caso dos eventos com alta luminosidade esta média se eleva para 155 ( $\pm 58, 6$ ) pontos. A diferença em número de pontos indica uma maior quantidade de ruído de fundo nos casos de alta luminosidade, onde um evento contém aproximadamente 23 eventos empilhados. Desta forma, em eventos de alta luminosidade, os empilhamentos podem ocasionar múltiplos elétrons, dificultando a localização do elétron atual, já que o de eventos anteriores ainda deixa rastro no detector.

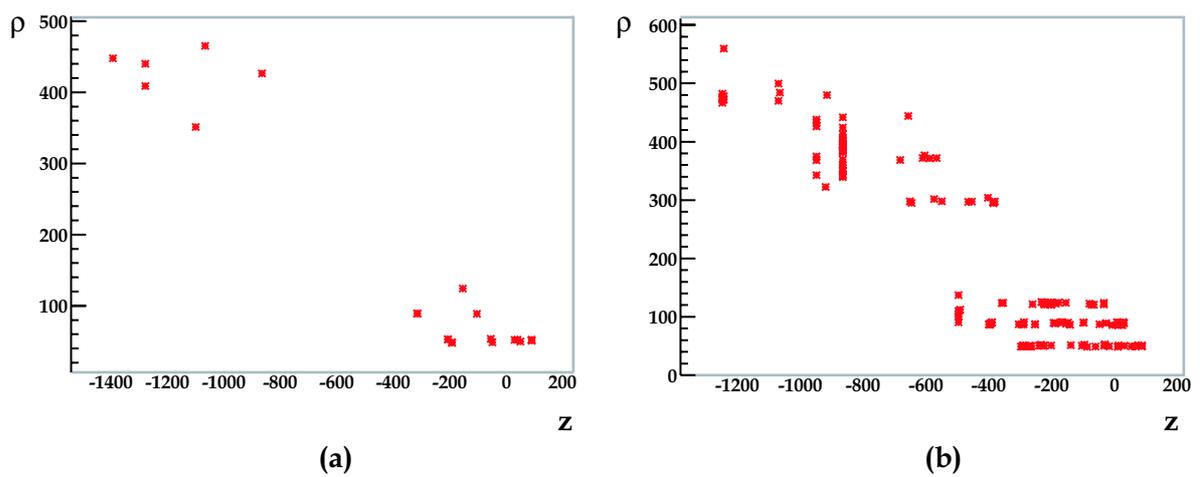


Figura 5.5: Exemplo de um elétron (a) com baixa luminosidade e (b) com alta luminosidade observados nos detectores de traço PIXEL e SCT.  $z$  é a direção do feixe e  $\rho$  é o raio das camadas dos detectores de traço.

# Capítulo 6

## O Sistema de Seleção de Regiões

Conforme abordado no capítulo anterior, o conceito de seleção de regiões foi utilizado para filtrar o ruído de fundo possibilitando trabalhar com a granularidade completa dos detectores. No âmbito do experimento ATLAS este conceito é mais amplo englobando todos seus detectores, sendo conhecido como regiões de interesse (RoI).

O mecanismo de determinação de regiões de interesse possibilita a decisão *online* de filtragem do HLT, uma vez que ele permite a redução significativa do volume inicial dos dados em uma colisão. Esta redução na informação a ser tratada pelo HLT deve ser realizada logo na entrada do sistema de filtragem. Desta forma, o mecanismo de seleção de regiões ocorre logo após a execução do primeiro nível de filtragem do ATLAS, consistindo na primeira ferramenta do sistema HLT.

Este capítulo aborda o mecanismo de funcionamento da ferramenta de seleção de regiões desenvolvida para o sistema de filtragem HLT. Conforme apresentado na Introdução, o desenvolvimento desta Tese está sendo realizado em colaboração com o experimento ATLAS no CERN. No período de setembro de 2002 à abril de 2003 foi realizado um estágio no CERN financiado pela CAPES. Durante este estágio foi desenvolvida a ferramenta de seleção de regiões (*Region Selector*), que atualmente é amplamente utilizada pelos grupos do HLT e de Reconstrução do ATLAS.

O Seleccionador de Regiões foi desenvolvido com a finalidade de traduzir regiões físicas de qualquer subdetector do ATLAS em Identificadores que nomeiam grupos de eventos de dados dentro de determinada granularidade.

## 6.1 Estratégia de Seleção e Princípios Gerais

A estratégia de seleção de eventos no HLT consiste na utilização de uma região geométrica, definida pelo primeiro nível de filtragem, para determinar uma Região de Interesse (RoI) que contém a assinatura de um objeto físico (elétron, fóton, etc.)[82].

O Seleccionador de Regiões possui o objetivo de traduzir uma região geométrica, *i.e.*, a região diretamente derivada da RoI do nível 1, em uma série de Identificadores *Offline*. A Figura 6.2 apresenta alguns exemplos de regiões. Normalmente estes identificadores representam um elemento do detector, chamado *DetectorElement Identifier*. Com base na lista destes Identificadores, um algoritmo obtém dados de eventos dentro de coleções intituladas pelo mesmo Identificador dentro do repositório de eventos TES (*Transient Event Store*). No nível 2, se estes Identificadores não estiverem armazenados dentro do TES, o sistema de busca de dados *StoreGate* solicita os dados provenientes do Coletor de Dados de Saída, chamados *Read-Out-Buffers Data Collector* (ROB).

Em alguns contextos, pode parecer mais coerente utilizar a seqüência na qual a decisão do nível 1 de *Trigger* é derivada a partir de dados de eventos *Online* ou identificadores de *hardware*. Estes indicadores poderiam, em princípio, ser utilizados para solicitar dados em eventos, ao invés de serem inseridos em camadas de abstração e envolvendo o Seleccionador de Regiões.

Existe uma variedade de motivações para utilizarmos camadas de abstrações envolvendo o Seleccionador de Regiões. A motivação principal é ganhar acesso a dados de eventos ocorridos em subdetectores que não participam da decisão de nível 1 de filtragem de forma uniforme e rápida. Uma motivação adicional inclui o acesso à RoI secundárias conforme a necessidade do algoritmo que encontra-se fora da RoI primária proporcionada pelo nível 1.

Embora Identificadores *Online* ROB sejam utilizados para requisição de dados, os eventos de dados são organizados e armazenados em coleções intituladas pelo Identificador *Offline*. Os eventos de dados, por exemplo, objetos de dados brutos (*Raw Data Objects* - RDO) e os objetos de entrada da reconstrução (*Reconstruction Input Objects* - RIO), são agregados em coleções correspondentes aos canais adjacentes dentro de um subdetector físico. Estas coleções encontram-se dentro de um

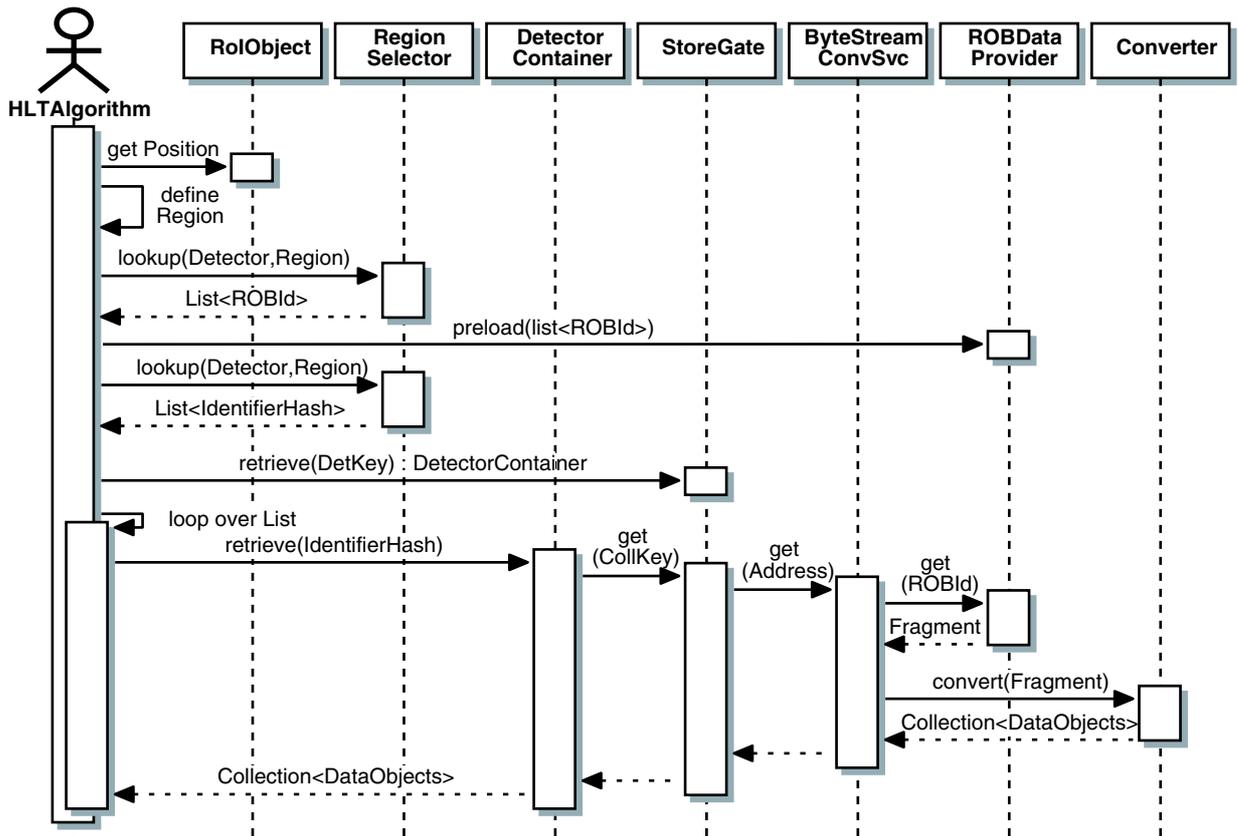


Figura 6.1: Modelo de acesso aos dados do PESA, referenciado como esquema Londres (*London scheme*), pelo qual os algoritmos do HLT solicitam dados de eventos. Uma solicitação de um determinado algoritmo do HLT ao selecionador de regiões consiste na primeira etapa do sistema de nível 2[7].

*IdentifiableContainer* (IDC) [7] com Identificadores correspondendo às unidades de agregação. Para a maioria dos subdetectores, o princípio de organização é aquele do *DetectorElement*.

No detector Pixel um *DetectorElement* é um módulo, equivalente a uma única camada de Sílica (*Silicon wafer*). Desta forma, temos 1744 *DetectorElements* no detector Pixel. Para o SCT, um *DetectorElement* é um lado de um módulo, equivalente a um par de camadas ligadas cujas tiras são orientadas em uma única direção; temos 8176 SCT *DetectorElements*. Para o TRT, um *DetectorElement* é um conjunto planar de tubos representando uma linha em um determinado raio do módulo dos tubos em um tambor e  $1/23$  na direção  $\eta\phi$  para um dado  $z$  de tubos; temos 19008 TRT *DetectorElements*.

Para os calorímetros, o conceito de *DetectorElement* não existe. Desta forma, o princípio de organização para os dados de um evento são os chamados Trigger Tower.

Dentro do espectrômetro Muon, para os MDT's, um *DetectorElement* é uma simples câmera MDT, onde há no máximo uma câmera simples MDT por estação, e tipicamente, uma câmera MDT possui duas multicamadas. O *DetectorElement* do RPC é o conjunto de componentes associados a um tambor da estação Muon; pode existir 0, 1 ou 2 conjuntos duplos RPC por estação e um conjunto duplo pode compreender 1,2 ou 4 duetos RPC. Um *DetectorElement* do TGC é uma estação; existem 24 estações dianteiras em forma de anel e mais 48 em estações laterais e quatro anéis em cada terminação lateral do detector ATLAS. Finalmente, para o detector CSC, um *DetectorElement* é um simples tambor, onde existe um tambor CSC por estação Muon.

## 6.2 Implementação e Projeto do Seleccionador de Regiões

O projeto do Seleccionador de Regiões pode ser separado em duas partes distintas: a fase de inicialização e a de execução. Um diagrama conceitual de seqüências é apresentado na Figura 6.3.

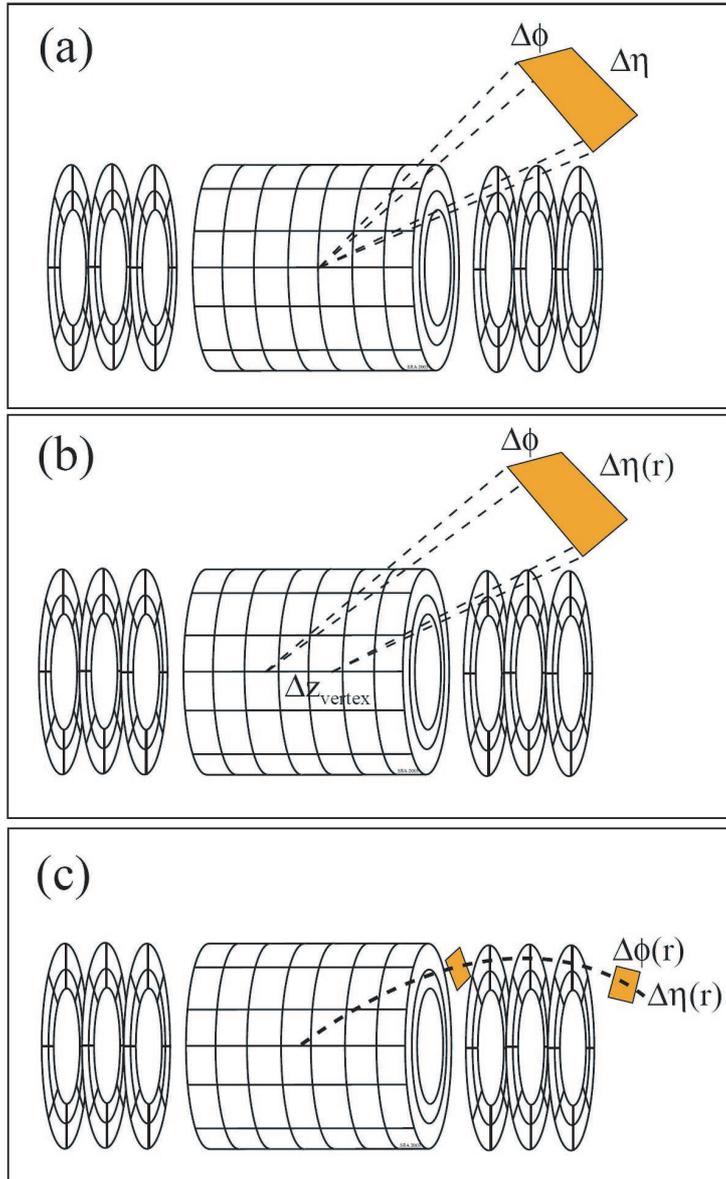


Figura 6.2: Exemplos de regiões. (a) apresenta uma simples extensão em  $\eta$  e  $\phi$  que não leva em conta a incerteza substancial na posição  $z$  do vertex primário devido ao espalhamento do feixe LHC. (b) apresenta uma extensão em  $\eta$  e  $\phi$  que leva em consideração o espalhamento do vertex  $z$ ; desta forma,  $\Delta\eta$  possui uma dependência radial. (c) apresenta uma caminho helicoidal, onde ambos  $\Delta\eta$  e  $\Delta\phi$  possuem dependência radial. Na presente implementação, (a) está implementado para os detectores externos; (b) está implementado para SCT e Pixels através de modificações nas extensões dos elementos detectores.

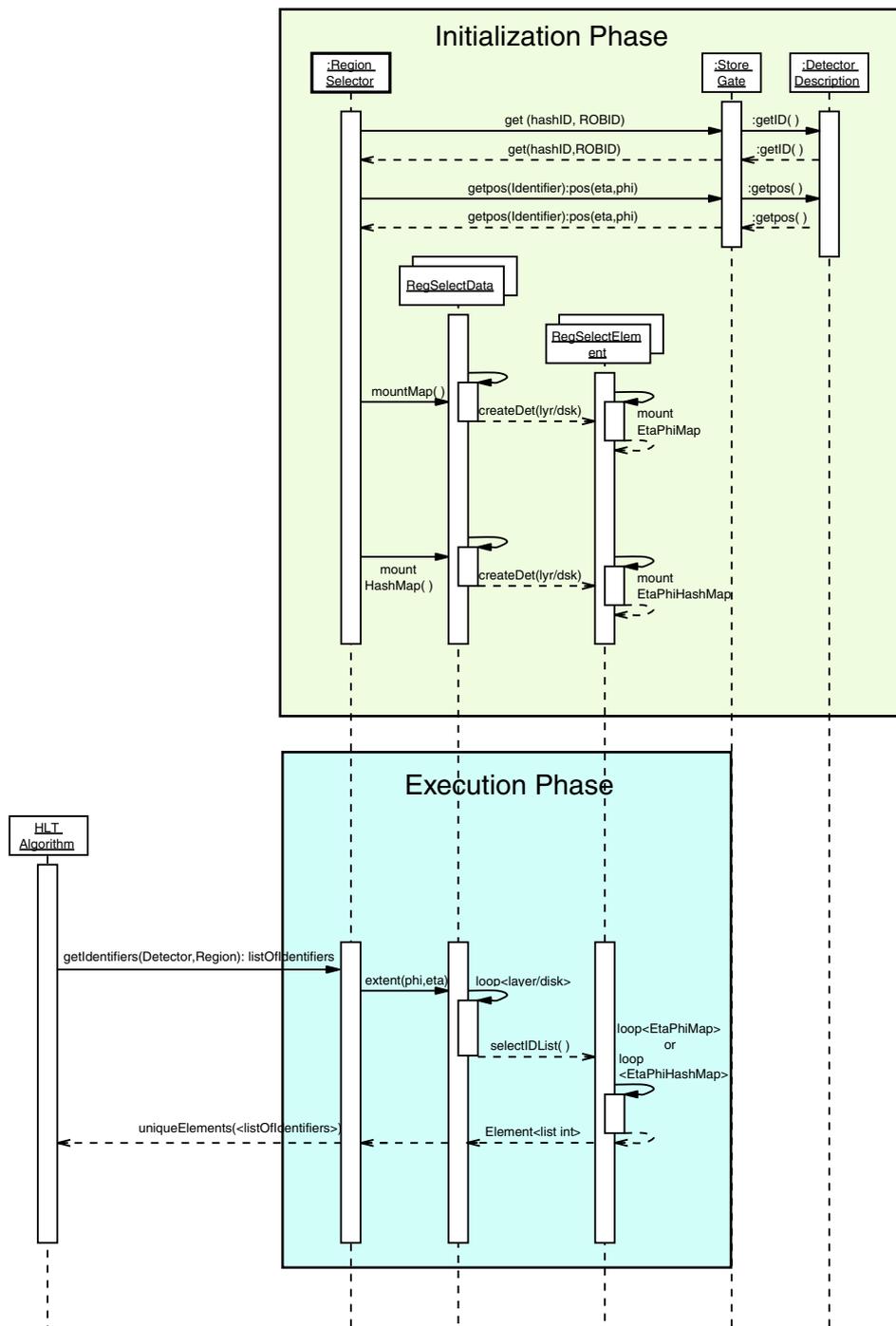


Figura 6.3: Ilustração do diagrama de seqüência conceitual distinguindo as duas fases do Seleccionador de Regiões.

Extended Identifier	IdentifierHash	$\phi_{min}$	$\phi_{max}$	$\eta_{min}$	$\eta_{max}$
2/2/-2/0/0/0/0	0	0,141628	0,265983	-1,47148	-1,11859
2/2/-2/0/0/0/1	1	0,136356	0,270412	-1,47293	-1,11859
2/2/-2/0/0/1/0	2	0,077387	0,236772	-1,73596	-1,31688
2/2/-2/0/0/1/1	3	0,070819	0,242117	-1,73833	-1,31350
2/2/-2/0/1/0/0	4	0,262459	0,386813	-1,46680	-1,11363
...	...	...	...	...	...

Tabela 6.1: Exemplo ilustrativo de um arquivo de dados ASCII do Seleccionador de Regiões contendo dados do *DetectorDescription*. Para subdetectores com múltiplas camadas, há a informação do raio para cada elemento. Dentro do mapa interno *EtaPhiMap*, somente *HashIdentifiers* são utilizados. Os *ExtendedIdentifiers*[83] são utilizados pelos programadores por motivos de verificação de informações.

O Seleccionador de Regiões utiliza a descrição dos detectores ATLAS durante o processo de inicialização. Na atual implementação, há arquivos de dados ASCII para cada subdetector que dão a informação necessária para montar um mapa interno de listas de Identificadores, referenciado como *EtaPhiMap*. Um exemplo ilustrativo de um arquivo de dados ASCII está representado na Tabela 6.1.

O *EtaPhiMap* existe para cada camada (ou disco) de um subdetector e é essencialmente um conjunto de vetores em  $\eta$  e  $\phi$ . A ordem de apresentação dos elementos nestes vetores é como àquela do arquivo de dados. Os elementos que compõem o *ExtendedIdentifier* possui a informação do endereço daquele determinado *HashIdentifier*, e são as peças chaves para a construção da estrutura de dados do Seleccionador de Regiões. De modo a termos os menores tempos de execução possíveis para selecionarmos regiões, a estrutura de armazenamento dos dados torna-se fundamental.

Diferentes métodos de construção de *EtaPhiMap* foram testados na busca da melhor performance de execução do algoritmo. O método escolhido foi aquele que divide os dados do detector com a menor granularidade possível, i.e., transforma cada camada e disco do detector em células independentes. Desta forma os dados do detector são subdivididos e o Seleccionador de Regiões possui a tarefa de escolher quais são as camadas e/ou discos responsáveis pela região de entrada. A ordem dos

elementos dentro das células indica o intervalo válido em  $\eta$  e  $\phi$  pra aquele *HashIdentifier*. Desta forma o *EtaPhiMap* trata o subdetector de uma maneira completamente genérica. A Figura 6.4 apresenta o *EtaPhiMap*, que ilustra a sobreposição de identificadores em determinadas regiões.

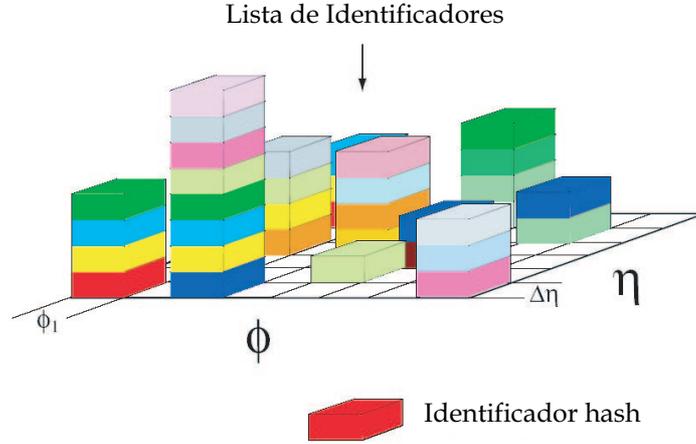


Figura 6.4: Ilustração do *EtaPhiMap*. Cada elemento pode conter identificadores repetidos com relação a seus vizinhos, devido a sobreposição dos componentes dos detectores.

As entradas da interface API do Seleccionador de Regiões são os subdetectores em consideração, i.e., PIXEL, SCT, TRT, LAR, TILE, MDT, RPC, CSC e TGC, e a extensão da região geométrica a ser considerada. Com a informação das camadas e/ou discos nas regiões, o algoritmo busca pelos *HashIdentifiers* somente nas células relevantes. Os *HashIdentifiers* apresentados na saída são uma lista ordenada e não repetidas aqueles identificadores dentro do intervalo requisitado. A Figura 6.5 apresenta de forma simplificada o diagrama de classes do Seleccionador de Regiões. Cada subdetector é um *RegSelectData*, e suas camadas e/ou discos são *RegSelectElements*. De fato, existe uma classe para lidar com detectores que possuem sobreposição de regiões, chamada *RegSelectData*, e outra para detectores sem sobreposição, como os Calorímetros e Múons, chamada *LArRegSelectData*. No momento, somente *LArRegSelectData* possui um mapa *EtaPhiMap* hash, possibilitando uma seleção mais rápida das regiões.

Através do conhecimento das camadas e discos que pertencem a região, o algoritmo busca os mapas  $\phi \rightarrow \text{IdentifierHash}$  que irão fornecer a seqüência de *IdentifierHash* relevantes na região  $\phi$ . O último passo do algoritmo consiste em

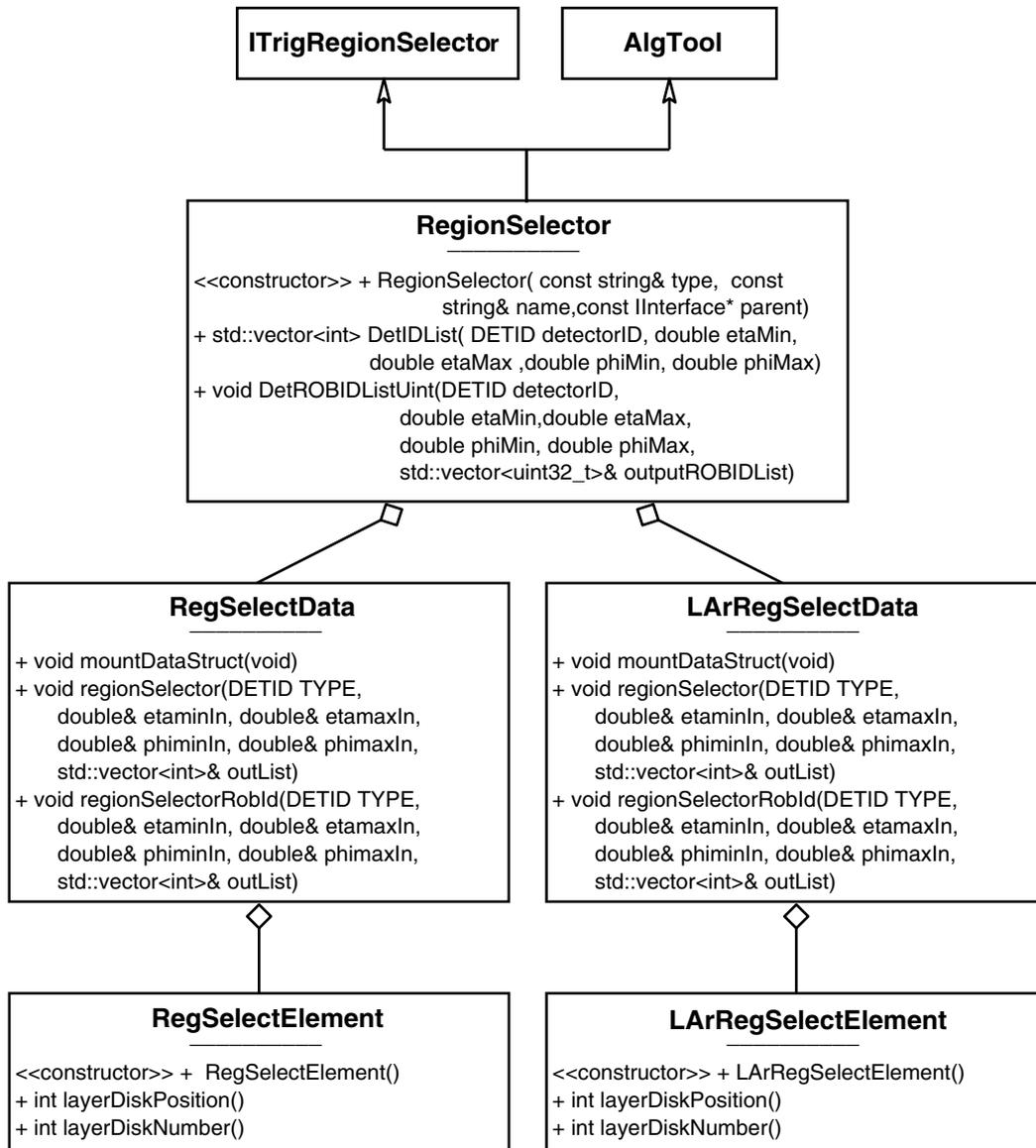


Figura 6.5: Diagrama simplificado de classes do Seleccionador de Regiões. Duas classes diferentes `Data` e `Element` formam os detectores, que por sua vez pode ter sobreposição `RegSElect...` ou não ter `LArRegSelect...`. Explicação mais detalhada encontra-se no texto.

validar cada `IdentifierHash` dentro do mapa `IdentifierHash`  $\rightarrow (\eta_{min}; \eta_{max})$ , de forma a assegurar saídas com elementos únicos.

O código contém muitas variações de classes e métodos que não estão no diagrama da Figura 6.5, para melhor compreensão. Interfaces ligeiramente diferentes são necessárias para descrever a região em detectores diferentes, como apresentam as Tabelas 6.2 e 6.3.

	Função	Entradas	Saída
1	<code>DetIDList</code>	<code>detectorID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;int&gt;</code>
2	<code>DetIDList</code>	<code>detectorID</code> , <code>sampling</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;int&gt;</code>
3	<code>DetIDList</code>	<code>detectorID</code> , <code>typeID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;int&gt;</code>
1	<code>DetHashIDList</code>	<code>detectorID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;IdentifierHash&gt;</code>
2	<code>DetHashIDList</code>	<code>detectorID</code> , <code>sampling</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;IdentifierHash&gt;</code>
3	<code>DetHashIDList</code>	<code>detectorID</code> , <code>typeID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;IdentifierHash&gt;</code>
1	<code>DetROBIDList</code>	<code>detectorID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;int&gt;</code>
2	<code>DetROBIDList</code>	<code>detectorID</code> , <code>sampling</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;int&gt;</code>
1	<code>DetROBIDListUint</code>	<code>detectorID</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;uint32 t&gt;</code>
2	<code>DetROBIDListUint</code>	<code>detectorID</code> , <code>sampling</code> , $\eta_{min}, \eta_{max}, \phi_{min}, \phi_{max}$	<code>vector&lt;uint32 t&gt;</code>

Tabela 6.2: Variações do método `RegionSelector`.

A Tabela 6.2 apresenta interfaces diferentes para descrever uma região nos diferentes detectores. A principal diferença está na saída, onde há *Offline Identifier-Hash* e *ROBIDs*. O termo `detectorID` é um *enum* para os identificadores (`PIXEL`,

Detectores	DetIDList	DetHashIDList	DetROBIDList	DetROBIDListUInt
LAr and Tile	1 and 2	1 and 2	1 and 2	1 and 2
Pixel, SCT and TRT	1	1	1	1
MDT, RPC and TGC	1 and 3	1 and 3	1	1

Tabela 6.3: Correspondências entre métodos e detectores.

SCT, TRT, LAR, TTEM, TTHEC, TILE, MDT, RPC, TGC), o valor `sampling` consiste em um número inteiro que enumera as camadas do Calorímetro e `typeID` é um `enum` que endereça as câmeras de Múons de forma separada (BIL, BIS, BML, BMS, BOL, BOS, BMF, BIR, BOF, BOG ).

O programa de seleção de regiões (`RegionSelector`) encontra-se no repositório *offline* de programas do ATLAS dentro de `DetectorDescription/RegionSelector`.

## 6.3 Estudo da Performance do Sistema

O principal objetivo do mecanismo de RoI está em reduzir o tempo necessário para obter dados de um evento pela rede através da dramática restrição do volume relevante do subdetector. No entanto, se o Seleccionador de Regiões levar muito tempo para executar essa tarefa, seu propósito seria completamente falho. É difícil, no entanto, quantificar exatamente qual a performance do sistema mais adequada, mas certamente a execução total do Seleccionador de Regiões deve ser menor que 1ms em uma máquina Pentium 4GHz.

O tempo médio de execução para recuperar uma região em cada subdetector em um intervalo de  $\Delta\eta \times \Delta\phi$  varia de acordo com a região selecionada, como mostra a Tabela 6.4. Existe ainda a tentativa de diminuição destes tempos, especialmente para subdetectores com elevado número de identificadores. A performance do sistema foi medida variando-se o intervalo em  $\eta$  e  $\phi$  em toda a extensão do detector. Através desse estudo podemos identificar dois aspectos importantes; a verificação dos tempos de execução com relação ao posicionamento do intervalo selecionado e a verificação da existência das saídas esperadas. A Figura 6.6 apresenta um gráfico

Identifier Hash timing in ms			
	$0.1 \times 0.1$	$0.2 \times 0.2$	$0.5 \times 0.5$
PIXEL	0.049( $\pm 0.012$ )	0.055( $\pm 0.014$ )	0.0758( $\pm 0.030$ )
SCT	0.101( $\pm 0.015$ )	0.119( $\pm 0.018$ )	0.164( $\pm 0.040$ )
TRT	0.581( $\pm 0.131$ )	0.643( $\pm 0.165$ )	0.856( $\pm 0.211$ )
LAR	0.024( $\pm 0.032$ )	0.036( $\pm 0.033$ )	0.088( $\pm 0.036$ )
TTEM	0.017( $\pm 0.002$ )	0.027( $\pm 0.006$ )	0.068( $\pm 0.004$ )
TTHEC	0.019( $\pm 0.001$ )	0.031( $\pm 0.002$ )	0.083( $\pm 0.003$ )
TILE	0.013( $\pm 0.006$ )	0.013( $\pm 0.012$ )	0.020( $\pm 0.028$ )
MDT	0.021( $\pm 0.008$ )	0.023( $\pm 0.011$ )	0.030( $\pm 0.024$ )
RPC	0.015( $\pm 0.005$ )	0.016( $\pm 0.006$ )	0.020( $\pm 0.008$ )
TGC	0.014( $\pm 0.003$ )	0.014( $\pm 0.004$ )	0.016( $\pm 0.004$ )

Tabela 6.4: Representação dos tempos médios de execução do algoritmo. Estes tempos foram medidos em um Pentium IV 2.7GHz.

em três dimensões, representado sobre a forma de imagem, que indica os tempos em milisegundos no subdetector Pixel para todos os possíveis intervalos de variação em  $\Delta\eta \times \Delta\phi$ .

A Figura 6.6(g) mostra que o tempo de execução possui um comportamento linear de acordo com o tamanho da região, e que o tempo de execução é constante dentro de um tamanho de região.

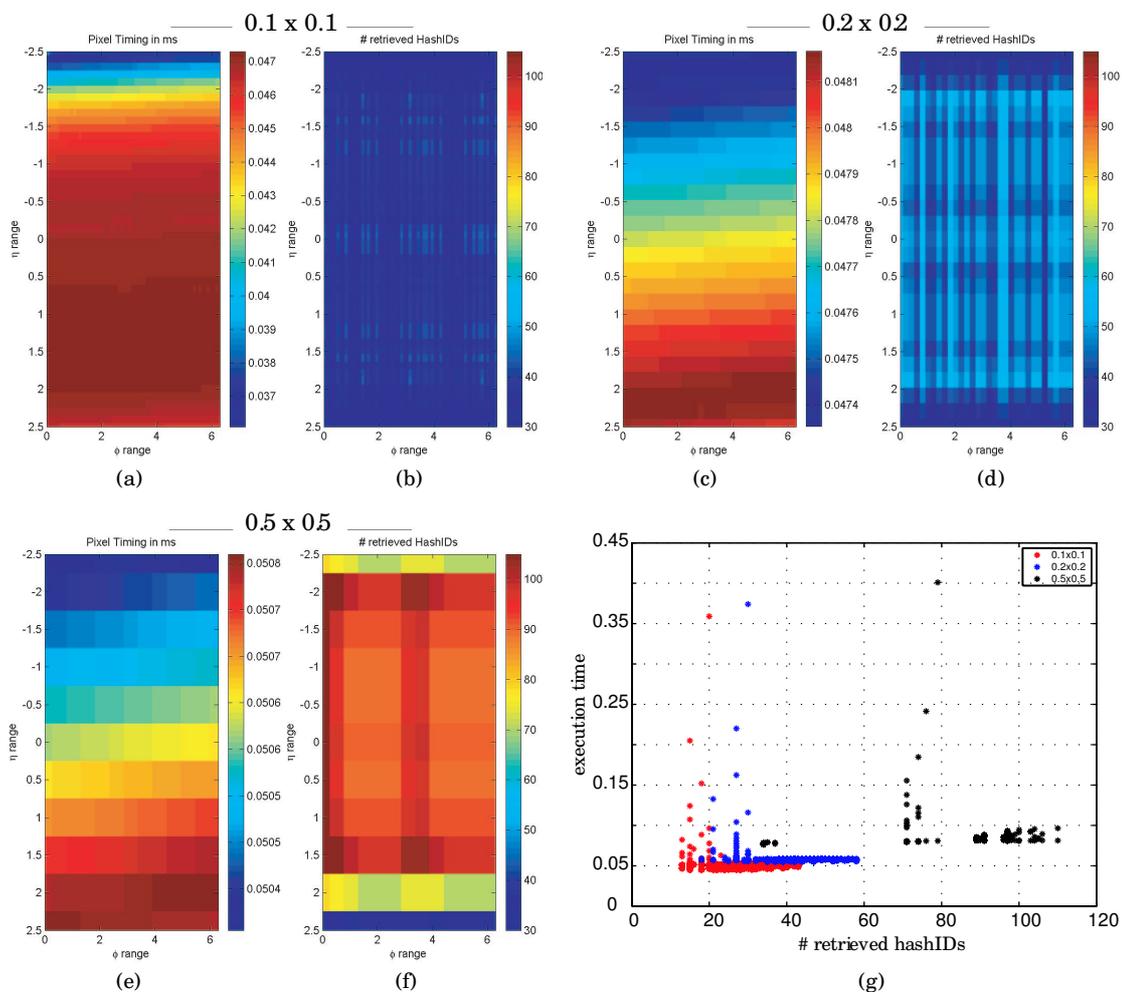


Figura 6.6: Tempos de execução para o subdetector Pixel em milisegundos. Os tempos de execução estão apresentados em (a), (c) e (e). O número de HashIDs de cada região está em (b), (d) e (f). O número de HashIDs em função do tempo de execução está representado em (g).

# Capítulo 7

## Apresentação e Análise do Algoritmo de Referência *ZFinder*

Conforme apresentado na Seção 3.2.1 este algoritmo é utilizado pelo segundo nível de *trigger* do ATLAS, como ponto de partida para o sistema de reconstrução de trajetórias *online*. O *ZFinder* utiliza uma aproximação linear entre  $z$  e  $\rho$ , considerando a trajetória de uma partícula carregada dentro de um campo magnético uniforme uma reta e não uma hélice.

Esta aproximação, apresentada pela Equação 3.18, reduz o custo computacional de implementação sem comprometer de forma significativa a eficiência de localização do vértice de colisão.

Este Capítulo apresenta o algoritmo *ZFinder*, abordando sua implementação, complexidade computacional e resultados de sua aplicação em eventos de elétrons individuais em alta e baixa luminosidade. A próxima Seção ilustra os passos de funcionamento do algoritmo para a obtenção do vértice primário, ou  $z_{vtx}$ . Em seguida, na Seção 7.2 é feita uma análise do cálculo de  $z_{vtx}$  utilizando pontos espaciais e não somente as coordenadas  $(\rho, z)$ . Por fim, a Seção 7.3 apresenta os resultados do *ZFinder* para os eventos de elétrons individuais com empilhamento a alta e baixa luminosidade. Estes eventos serão utilizados pelos algoritmos de reconstrução de vértices por transformada de Hough e os resultados serão comparados.

## 7.1 Algoritmo para o Cálculo de $z_{vtx}$

O valor  $z_{vtx}$  é comum a todas as trajetórias originadas de um mesmo vértice primário. Desta forma, é relevante a obtenção deste valor como primeiro passo de um sistema de reconstrução. Isto é, utilizar a informação de muitas trajetórias para calcular o valor de um parâmetro comum, o  $z_{vtx}$ .

O princípio deste algoritmo consiste em utilizar a Equação 3.18 com pares de pontos que estão em camadas distintas dos detectores, calcular a interseção com o eixo  $z$  e armazenar estes valores em um histograma comum, chamado daqui por diante de histograma- $z$ . Os pares de pontos que forem originados de uma mesma trajetória formarão um pico em torno de  $z_{vtx}$  no histograma- $z$ .

Caso todos os pares de pontos fossem considerados, qualquer pico no histograma- $z$  seria disfarçado pelo ruído combinatorial[84]. Em adição, em alta luminosidade, como mencionado em Drevermann [85], existem muitas interações  $pp$  ao longo do eixo  $z$ , mas somente o  $z_{vtx}$  de interesse corresponde à interação de mais alto momento transversal ( $p_T$ ), *i.e.*, a interação que representa a maior parte das trajetórias com alto  $p_T$  em um evento. Trajetórias com elevado  $p_T$  curvam-se pouco na direção do plano  $xy$ , desta forma, proporcionando pontos com quase o mesmo  $\phi$ . De forma a realçar a contribuição dos pares de pontos oriundos das trajetórias de alto  $p_T$  e, ao mesmo tempo, reduzir o ruído combinatorial, somente os pontos dentro de pequenas tiras de  $\phi$  são combinados. Portanto, o algoritmo segue os seguintes passos:

- Os pontos são divididos em conjuntos por tiras em  $\phi$  e por camada;
- Em cada tira de  $\phi$ , a interseção com o eixo  $z$  é calculada utilizando a Equação 3.18 para cada par de pontos de diferentes camadas. Caso o valor de  $z$  da interseção estiver dentro da faixa de variação do histograma- $z$ , este é armazenado no histograma. De modo a evitar efeitos de borda, e levar em consideração somente a medida da incerteza de medida de  $\phi$ , os pontos de uma dada tira de  $\phi$  é também combinado com suas duas tiras vizinhas;
- A posição  $z_{vtx}$  é calculada como a média ponderada de três bins consecutivos àquele de maior contagem no histograma- $z$ .

A estimação da redução do ruído combinacional é realizada pela divisão do número de pontos no evento em tiras de  $\phi$ . Para  $N_{sp}$  pontos, o número de todas as possíveis combinações entre seus pares é dado pela Equação 7.1.

$$N_{pares}^{max} = N_{sp} \cdot (N_{sp} - 1) / 2 \cong N_{sp}^2 / 2 \quad (7.1)$$

Caso os pontos estejam divididos em  $M$  conjuntos igualmente distribuídos, e somente os pontos dentro de cada par sejam combinados, o número total de pares fica como na Equação 7.2.

$$N_{pares} \cong M \cdot (N_{sp} / M)^2 / 2 \cong N_{pares}^{max} / M \quad (7.2)$$

Desta forma, há a redução pelo fator  $M$  do ruído combinacional. De forma correspondente, uma redução similar é obtida no tempo total de execução do algoritmo, uma vez que a perda maior de tempo está na combinação entre pontos e não na arrumação dos pontos em tiras em  $\phi$  e por camadas, nem no armazenamento de  $z$  no histograma.

O algoritmo *ZFinder* é conceitualmente simples e somente poucos parâmetros são necessários ajustar para otimizar sua eficiência. Estes parâmetros são:

- A largura das tiras em  $\phi$ : esta largura depende do  $p_T$  da trajetória esperada. Quanto menor as tiras em  $\phi$ , melhor para trajetórias de alto  $p_T$ ;
- A largura do bin do histograma- $z$ : o valor ótimo está na ordem de resolução de  $z_{vtx}$ ;
- O número de camadas a serem utilizadas. Caso muitas trajetórias com alto  $p_T$  estejam presentes, é suficiente restringir a combinação dos pontos ao detector PIXEL.

A Figura 7.1 apresenta um histograma- $z$  de um evento de baixa luminosidade com alto  $p_T$  e com  $z_{vtx} = -1,45\text{cm}$ . É possível notar que o ruído de fundo é baixo, ilustrando que o algoritmo não é sensível a combinação entre pontos para eventos de baixa luminosidade.

A Figura 7.2 apresenta o histograma- $z$  para um evento de alta luminosidade e alto  $p_T$ , com  $z_{vtx} = -7,55\text{cm}$ . Nota-se um pico na posição correta, mas a relação sinal ruído é bem baixa.

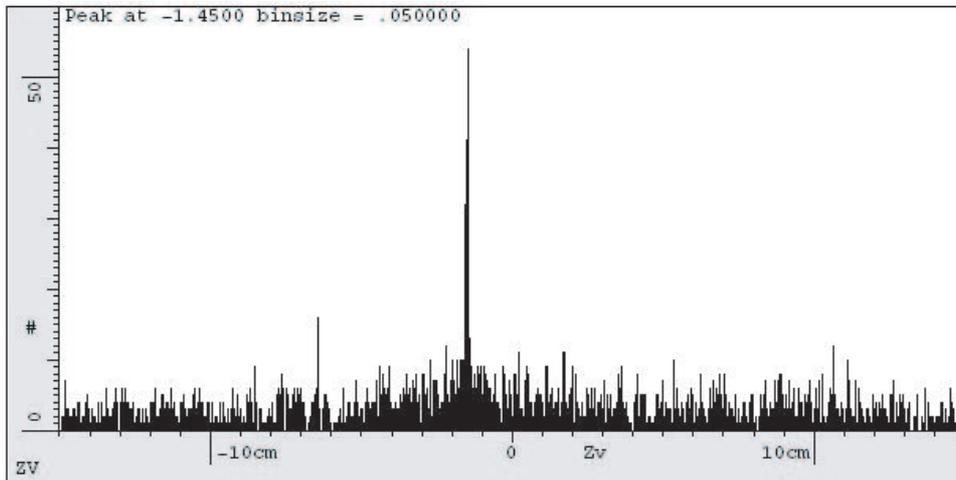


Figura 7.1: Histograma- $z$  para um evento de baixa luminosidade e alto  $p_T$ , sem empilhamento.

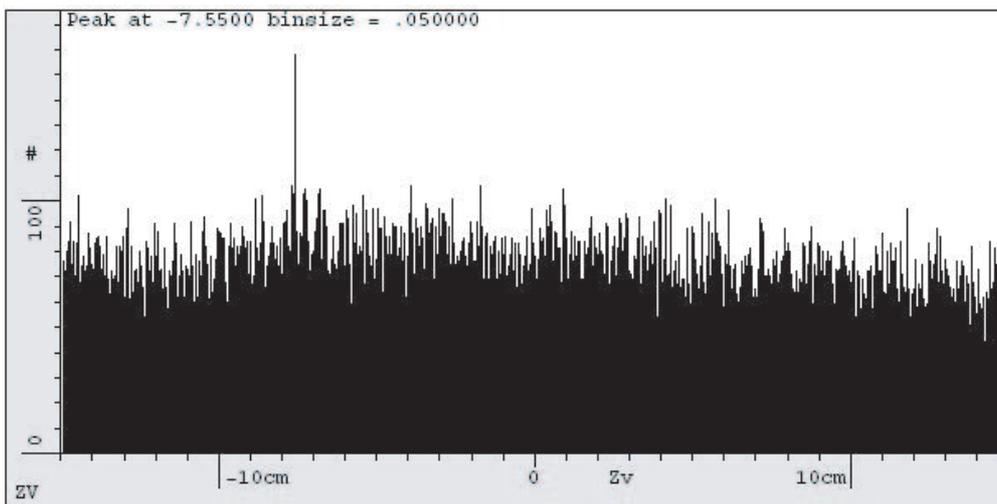


Figura 7.2: Histograma- $z$  para um evento de alta luminosidade e alto  $p_T$ , sem empilhamento.

## 7.2 Cálculo de $z_{vtx}$ Utilizando Trios de Pontos Tridimensionais

De modo a aumentar a relação sinal ruído para eventos de alta luminosidade, uma extensão do algoritmo descrito na Seção anterior foi desenvolvida. O  $z_{vtx}$  calculado a partir de um par de pontos nas camadas do detector PIXEL somente será aceito no histograma- $z$  caso haja um terceiro ponto em uma camada acima que forme o trio de pontos consistente com as Equações 3.15 e 3.17. O critério de consistência é o seguinte:

$$|z_{vtx}^{12} - z_{vtx}^{23}| = \Delta z_{vtx} \quad (7.3)$$

$$|C_{12} - C_{23}| = \Delta C \quad (7.4)$$

$$|G'_{12} - G'_{23}| = \Delta G \quad (7.5)$$

$$(7.6)$$

onde  $C_{nm}$ ,  $G'_{nm}$  e  $z_{vtx}^{nm}$  são calculados pelas Equações 3.15 e 3.17 para dois pontos  $n$  e  $m$ . Essencialmente o critério assume que três pontos serão originados da mesma trajetória. Caso  $N$  pontos de camadas mais externas produzirem um trio colinear com os pontos 1 e 2,  $z_{vtx}^{12}$  será armazenado no histograma- $z$   $N$  vezes.

O histograma- $z$  para eventos de alta luminosidade utilizando trio de pontos encontra-se representado na Figura 7.3. O nível de ruído reduziu bastante e o pico  $z_{vtx} = -7,5\text{cm}$  encontra-se predominante.

A divisão dos pontos em tiras de  $\phi$ , como na Seção anterior, reduz drasticamente as combinações. Para  $N_{sp}$  pontos, o número total de trios possíveis está descrito na Equação 7.7

$$N_{trios}^{max} = N_{sp} \cdot (N_{sp} - 1) \cdot (N_{sp} - 1) / 6 \cong N_{sp}^3 / 6 \quad (7.7)$$

enquanto que dividindo por  $M$  conjuntos iguais e realizando a combinação após a separação em trios há uma redução de  $M^2$ , como mostra a Equação 7.8.

$$N_{trios} \cong M \cdot (N_{sp}/M)^3 / 6 \cong N_{trios}^{max} / M^2 \quad (7.8)$$

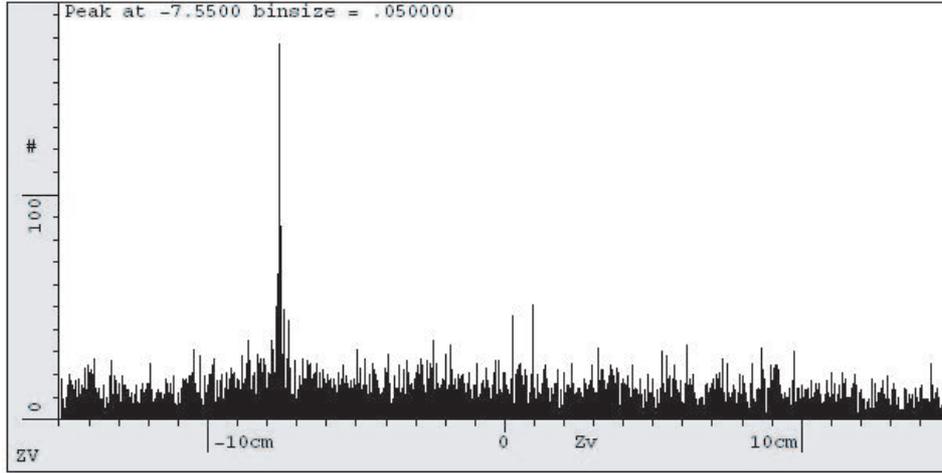


Figura 7.3: Histograma- $z$  para um evento de alta luminosidade e alto  $p_T$ , utilizando trio de pontos para formar o histograma.

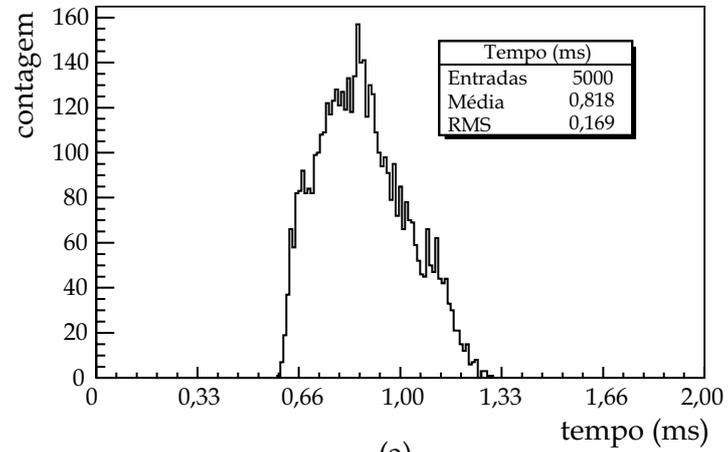
Na prática, os pontos são divididos em tiras de  $\phi$  de mesma largura. Como a distribuição de pontos não é uniforme as diferentes tiras de  $\phi$  não contêm o mesmo número de pontos. Portanto, o ganho real nas combinações é menor que  $M^2$ .

### 7.3 Resultados do Algoritmo *ZFinder* em Eventos de Elétrons Individuais

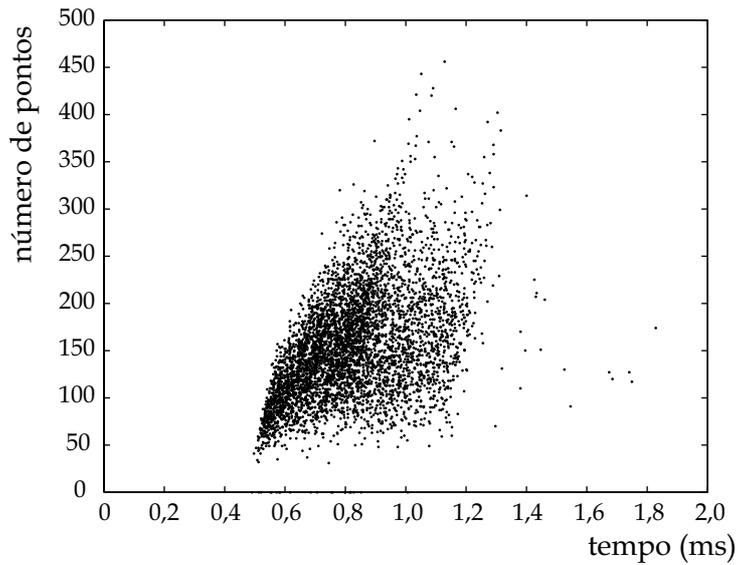
Os gráficos das Figuras 7.4, 7.5 e 7.6 apresentam as respostas de tempo de processamento e eficiência. Nos gráficos de resposta de tempo é possível observar o comportamento do tempo de processamento com relação ao número de pontos em um evento. Há uma tendência linear no tempo de execução, apesar da grande dispersão dos pontos no gráfico.

A Figura 7.4(a) apresenta o histograma do tempo de execução de amostras de alta luminosidade. A média está em 0,818 ms com um valor RMS de 0,169 ms. A Figura 7.4(b) apresenta o gráfico do número de pontos com relação ao tempo de execução. É observado um comportamento linear do tempo de processamento com relação ao número de pontos em um evento.

Em baixa luminosidade o tempo de execução possui o mesmo comportamento observado nas amostras de alta luminosidade. O histograma da Figura 7.5(a) apresenta os tempos de execução do *ZFinder* para amostras de baixa luminosidade. A



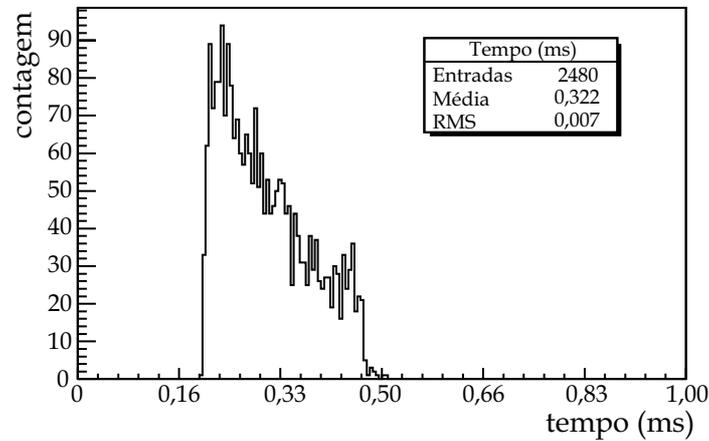
(a)



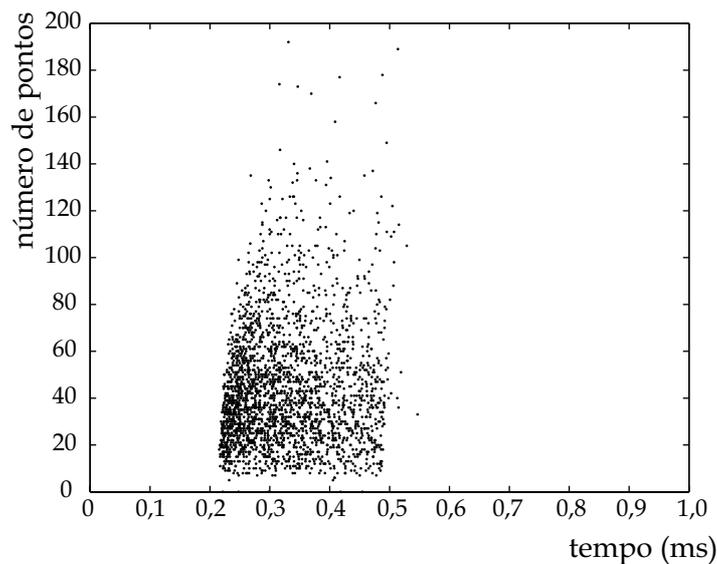
(b)

Figura 7.4: Tempos de execução do *ZFinder* para eventos de alta luminosidade. (a) histograma do tempo de execução; (b) gráfico do tempo de execução com relação ao número de pontos nos eventos.

média do tempo de execução encontra-se em 0,322 ms com um valor RMS de 0,007 ms. A Figura 7.5(b) apresenta o gráfico do tempo de execução com relação ao número de pontos em um evento. Também é observado um comportamento linear, apesar da dispersão dos pontos.



(a)



(b)

Figura 7.5: Tempos de execução do *ZFinder* para eventos de baixa luminosidade. (a) histograma do tempo de execução; (b) gráfico do tempo de execução com relação ao número de pontos nos eventos.

A Figura 7.6 apresenta a diferença entre os valores dos vértices obtidos pelo *ZFinder* ( $z_{otx}$ ) e os valores simulados, ou *true* ( $z_{true}$ ). Quanto mais próxima de um delta de Dirac melhor consiste na eficiência do sistema de localização. O valor médio do histograma para amostras de baixa luminosidade consiste em 0,04928 mm com

um RMS de 0,3861 mm. Para alta luminosidade o valor médio é de 0,03679 mm com um RMS de 0,3755 mm. A eficiência de acerto para eventos de baixa luminosidade consiste em 2342/2480 (94,43%), e para eventos de alta luminosidade consiste em 4356/4942 ou 4407/5000 (88,14%).

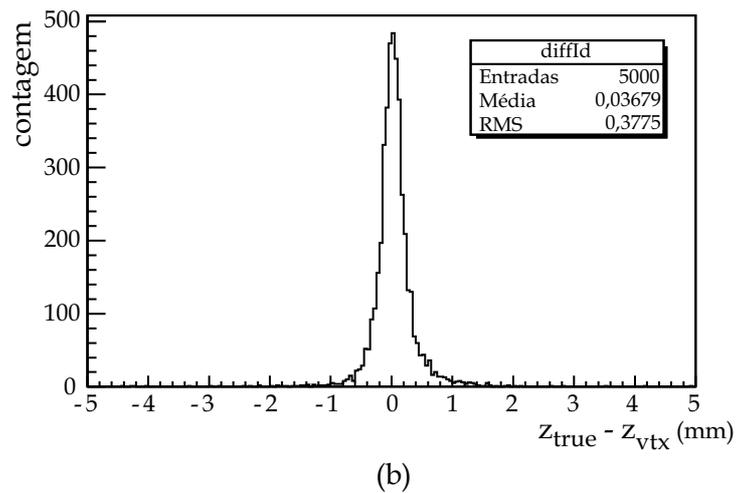
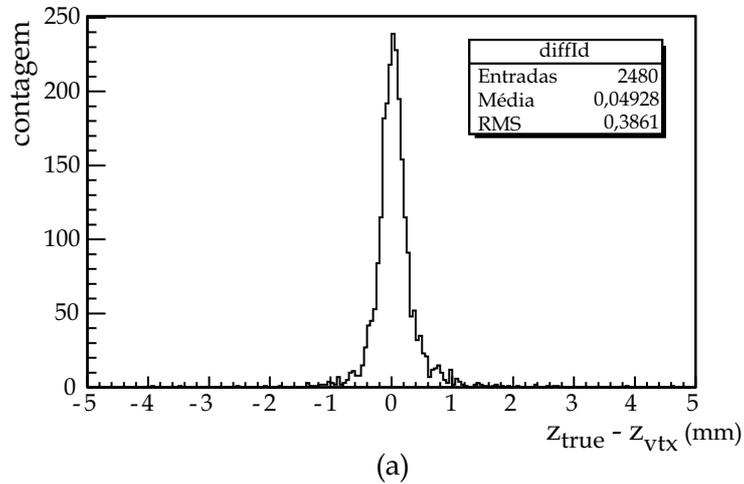


Figura 7.6: Diferença entre os valores de simulação ( $z_{true}$ ) e os valores obtidos pelo *ZFinder* ( $z_{vtx}$ ) para amostras de baixa (a) e alta (b) luminosidade.

# Capítulo 8

## Sistema de Localização de Vértices Primários por Transformada de Hough

O sistema de reconstrução de vértices primários de regiões de interesse pode ser dividido basicamente em dois blocos principais, que são: a seleção dos pontos e a localização por transformada de Hough. Este Capítulo apresenta os conceitos de implementação destas duas etapas do sistema.

A seleção dos pontos possui duas finalidades. A primeira consiste em reduzir o esforço computacional da etapa posterior, *i.e.*, a de localização. A segunda finalidade está em melhorar a eficiência do sistema, como será apresentada no Capítulo 9. Nessa tese foram realizadas três abordagens distintas de métodos de seleção tridimensionais, são elas: cone, raio de atuação em  $\phi \times z$  e vizinhança em  $\phi$ . A Seção 8.2 apresenta os detalhes destas abordagens, especificando as suas características e os seus fluxogramas.

O segundo bloco principal, o da transformada de Hough, realiza a etapa de localização do vértice primário da região de interesse, recebendo na sua entrada os dados filtrados. A Seção 8.3 tem por objetivo apresentar a aplicação dos conceitos de transformada de Hough, apresentados no Capítulo 4, na tarefa de reconstrução do vértice primário. Um total de três variações de transformada foi testado. A primeira consiste na transformada de Hough Local, a segunda é a transformada de Hough Adaptativa e a terceira consiste na transformada de Hough Morfológica.

Detalhes sobre todas as abordagens especificando as suas características e os seus fluxogramas de implementação são apresentados na Seção 8.3.

A Figura 8.1 apresenta o diagrama de blocos simplificado do sistema de reconstrução por transformada de Hough. Nesse diagrama estão representados três blocos, que são: o módulo de leitura dos eventos; o módulo de seleção de pontos; e o módulo de localização de vértices primários. Dentro do módulo de localização de vértices primários além dos blocos de transformada de Hough existem os blocos de escolha de pontos e condição de parada. Os detalhes de funcionamento de cada um desses blocos será visto na Seção 8.3.

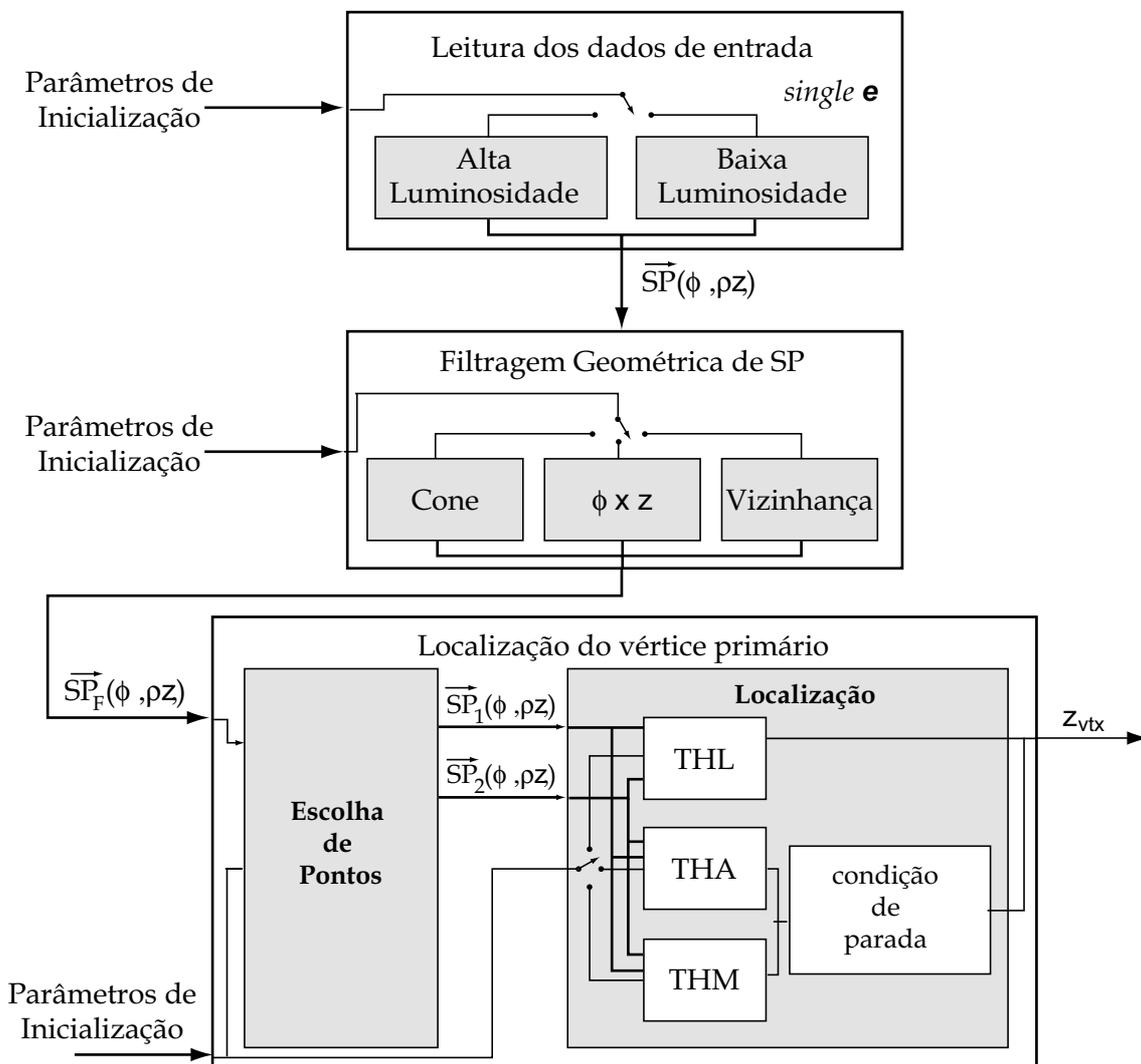


Figura 8.1: Diagrama de blocos simplificado do sistema de reconstrução de vértices primários por transformada de Hough.

## 8.1 Os Dados de Entrada

Conforme apresentado anteriormente, os dados de entrada do processo de reconstrução correspondem à imagem volumétrica dos detectores de traço. Essa imagem consiste em uma lista encadeada de pontos tridimensionais  $(\phi, \rho, z)$  contendo os “*voxels* ativos”<sup>1</sup> e a informação de sua localização nos detectores, *i.e.*, em qual camada ou disco os *voxels* se encontram. Esta estrutura de pontos tridimensionais é comumente referenciada por *Space Points* ou SP.

A lista SP $(\phi, \rho, z)$  é utilizada na abordagem de reconstrução de vértices por ajuste linear de retas na projeção  $\rho \times z$ , algoritmo *ZFinder*[49]. A Seção 3.2.1 demonstrou a aproximação da equação de hélices para retas nesta projeção. Da mesma forma que o algoritmo *ZFinder*, os métodos de transformada de Hough irão atuar como ajuste linear da lista de SP $(\phi, \rho, z)$  na projeção  $\rho \times z$ .

O *ZFinder* localiza o vértice de colisão com precisão de micrômetros e com  $\approx 0,8ms$  de tempo de processamento<sup>2</sup>. O algoritmo divide a região de interesse (RoI) de entrada em faixas de  $\phi$ . Cada par de pontos de camadas diferentes é utilizado para calcular a extrapolação linear em  $z$ . Os valores de  $z$  encontrados são acumulados em um histograma, sendo que o valor máximo deste histograma corresponde ao vértice primário da colisão.

O procedimento de reconstrução de vértices por transformada de Hough segue o mesmo princípio do *ZFinder*, *i.e.*, considera como linear a formação dos pontos na projeção  $\rho \times z$ . Desta forma, a transformada de Hough busca a origem da colisão através do ajuste linear da lista de pontos  $(\phi, \rho, z)$  nesta direção. A eficiência da reconstrução consiste em obter valores absolutos da subtração entre o vértice simulado (*true*) e o vértice obtido menores que 1 mm.

Uma vez que o sistema de reconstrução de vértices primários requer o menor esforço computacional possível para realizar a sua tarefa, os diversos algoritmos implementados foram otimizados de forma a atingir esse menor tempo sem alterar a eficiência ou precisão de reconstrução de vértices já obtida pelo *ZFinder*.

---

<sup>1</sup> *Voxel* é a abreviação de *volume pixel*, *i.e.*, é o equivalente ao pixel porém contendo três dimensões.

<sup>2</sup> Este tempo foi medido em amostras de alta luminosidade e escalado para uma máquina de 4 GHz e medido no ambiente de desenvolvimento ATHENA.

## 8.2 Abordagem de Seleção de Pontos

Os dados de entrada do sistema foram obtidos através do mecanismo de filtragem do LVL1 do ATLAS e do sistema de seleção de regiões (Capítulo 6). Os pontos tridimensionais estão representados em um sistema de coordenadas cilíndricas. A seleção dos pontos de entrada, por sua vez, utiliza *a priori* a informação de buscas por retas na direção  $\rho \times z$ .

Três abordagens distintas foram testadas, são elas: cone com base em  $(\phi, z)$ ; raio de atuação em  $\phi \times z$  e; vizinhança em  $\phi$ . Todas estas linhas de ação representam métodos de seleção geométricos, que levam em consideração os pontos tridimensionais de entrada.

### 8.2.1 Seleção por Cone

A seleção por cone consiste numa seleção geométrica em forma de cone com base no plano  $\phi \times z$ . A altura e a inclinação do cone são determinadas pelo ponto de vértice. No caso específico de localização de vértices primários para uma dada região de interesse do ATLAS, o vértice do cone pode ser representado pelo ponto médio de absorção de energia nos calorímetros.

Os calorímetros são os detectores do ATLAS que medem a energia incidente das partículas por meio de sua absorção. Eles são altamente segmentados, possuindo alta resolução. O perfil de deposição da energia no detetor depende do tipo de partícula que com ele interage. Com isto, os calorímetros são usados em sistemas *online* de validação de eventos, aproveitando da sua alta eficiência de discriminação e de sua velocidade de processamento. Neste caso, a informação geométrica do ponto médio de absorção de energia é utilizada como vértice da seleção por cone. O sistema desenvolvido por Stathis, apresentado na Seção 3.2.2 calcula esse ponto médio de absorção.

Uma vez que os dados de entrada disponíveis correspondem aos detectores internos do ATLAS, eles não contêm a informação dos calorímetros. Desta forma, de modo a obter o ponto de vértice para a construção do método de seleção por cone foram utilizadas as informações de simulações disponíveis.

Os valores do vértice e da inclinação do cone são simulados através dos valores

de  $z_{true}$  e do coeficiente angular da reta, que determina o valor do vértice primário de colisão. Estes últimos valores estão disponíveis nos dados simulados de entrada. O raio do cone é fixo possuindo valor igual a  $1\text{ mm}$ , correspondendo ao valor de tolerância aceitável para a determinação do vértice primário de colisão.

A Figura 8.2 representa um exemplo do emprego do método de seleção por cone aplicado a um evento de elétrons individuais à alta luminosidade. Neste caso o vértice do cone foi calculado através dos parâmetros de simulação. São apresentadas as projeções  $\rho \times z$  e  $\phi \times z$  de um evento de elétron individual a alta luminosidade e com empilhamento. As Figuras 8.2(a) e 8.2(b) apresentam o gráfico do evento original e as Figuras 8.2(c) e 8.2(d) o evento após a aplicação do método de seleção por cone. É possível observar que a maioria dos pontos foram descartados após a seleção.

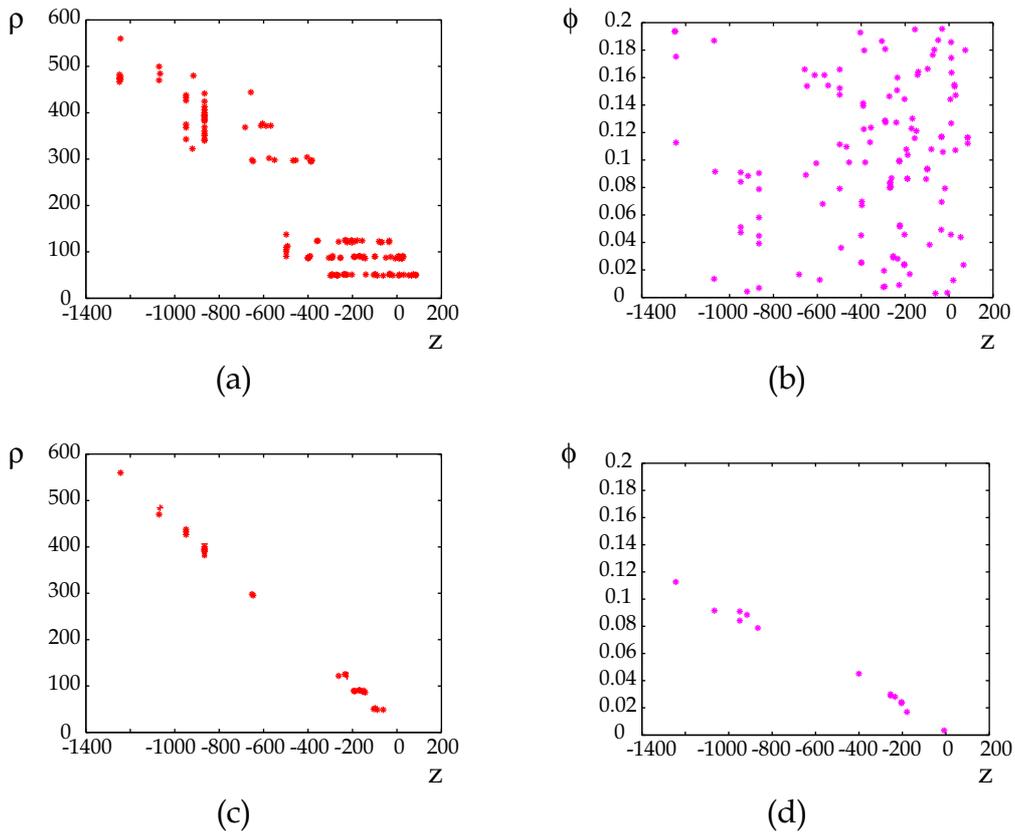


Figura 8.2: Exemplo do método de seleção por Cone aplicado a um evento de alta luminosidade. (a) projeção  $(\rho, z)$  do evento; (b) projeção  $(\phi, z)$ ; (c) projeção  $(\rho, z)$  após aplicação da seleção; (d) projeção  $(\phi, z)$  após aplicação da seleção.

## 8.2.2 Seleção por Raio de Atuação em $\phi \times z$

O princípio do método de seleção por raio de atuação consiste em estabelecer um alcance de combinações para os pontos tridimensionais do evento de entrada. O raio é estabelecido na direção  $\phi \times z$  e somente os pontos dentro deste raio são utilizados nas combinações dois a dois da transformada de Hough. Desta forma, o ruído ocasionado pela combinação indiscriminada de pontos é reduzido.

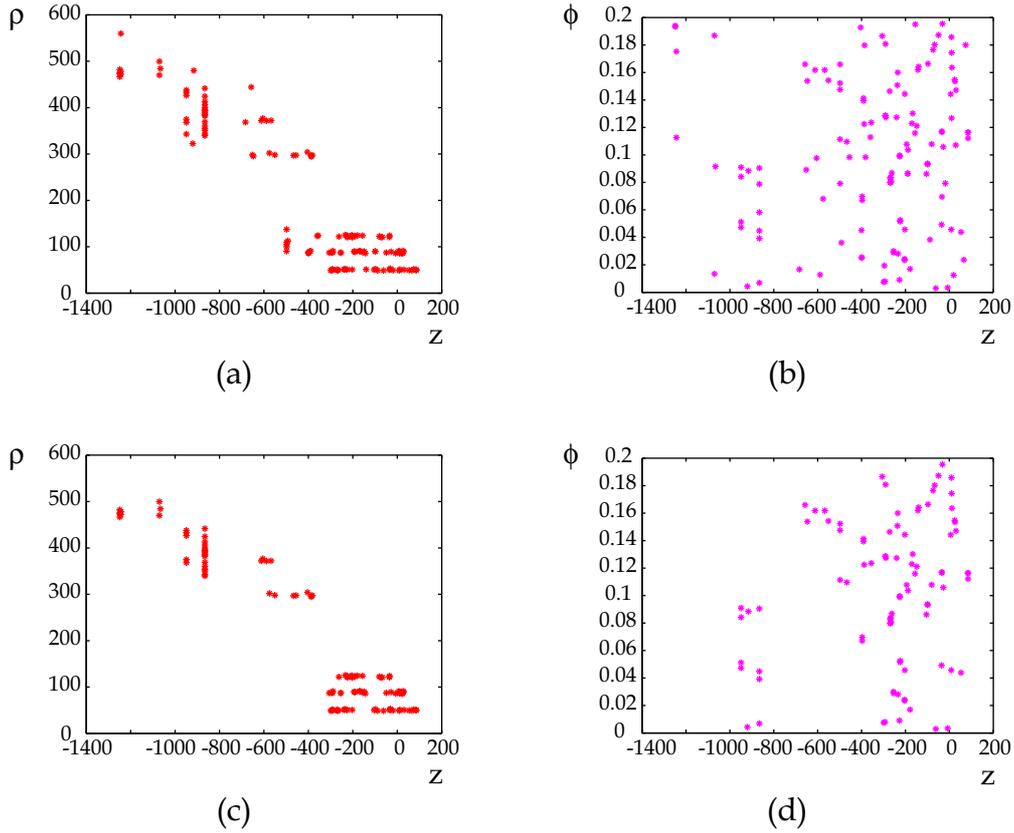


Figura 8.3: Exemplo do método de seleção por raio de atuação em  $\phi \times z$ . (a) evento original; (b) ilustração do cilindro de atuação ( $\phi, z$ ); (c) evento filtrado; (d) projeção ( $\phi, z$ ) após aplicação da seleção.

A Figura 8.3 apresenta um exemplo da aplicação do método de seleção por raio de atuação em um evento de alta luminosidade. Nesta seleção não há uma diminuição significativa de pontos, como é observado comparando as Figuras 8.3(a) e 8.3(b) com as Figuras 8.3(c) e 8.3(d). Isto se deve ao fato do raio de atuação ser largo. Quanto mais largo o raio de atuação mais combinação entre pontos são permitidas.

Esse método de seleção funciona da seguinte forma: um ponto da camada

mais interna (ponto de menor  $\rho$ ) é selecionado; em seguida o raio de atuação é aplicado a este ponto. São selecionados todos os pontos das camadas superiores para serem utilizados na transformada de Hough. Em seguida o mesmo processo é repetido para um outro ponto dessa mesma camada mais interna do detector. Quando todos os pontos da camada mais interna foram utilizados o mesmo procedimento é realizado para a camada imediatamente acima. Os gráficos apresentados nas Figuras 8.3(c) e 8.3(d) correspondem aos pontos que permaneceram no evento após o processo de seleção por raio de atuação e  $\phi \times z$ .

### 8.2.3 Seleção por Vizinhaça em $\phi$

O método de seleção por vizinhaça em  $\phi$  possui um conceito bastante similar ao método de seleção por raio de atuação em  $\phi \times z$ . Este método é o utilizado no algoritmo *ZFinder*. O método de seleção por vizinhaça em  $\phi$  também utiliza um raio de atuação em  $\phi \times z$ . A diferença para o método de seleção por raio de atuação em  $\phi \times z$  é que neste o raio de atuação é discreto e somente os pontos vizinhos são selecionados.

Desta forma, este método utiliza um mapa bidimensional discreto em  $\phi$  e  $z$ . Os pontos tridimensionais de entrada são dispostos neste mapa, onde a informação de  $\rho$  não é considerada. A dimensão do mapa é estabelecida pelo intervalo de variação de  $\phi$  e  $z$  dos pontos de entrada, e o número de divisões é proporcional a um parâmetro físico chamado pseudorapidez média  $\mu_\eta$ . Todos os pontos que possuem vizinhos neste mapa bidimensional  $(\phi, z)$  são considerados relevantes para a determinação do vértice primário de colisão.

A Figura 8.4 apresenta um exemplo deste método. A Figura 8.4(a) apresenta as projeções  $\rho \times z$  e  $\phi \times z$  do evento original de entrada a alta luminosidade. O espaço discreto  $z \times \phi$  é apresentado na Figura 8.4(b). Todos os pontos representados na Figura 8.4(a) estão em 8.4(b), onde agora encontram-se discretizados. A Figura 8.4(c) apresenta o mapa  $z \times \phi$  após a seleção dos pontos. É possível observar que todos os pontos que não possuíam vizinhos na camada imediatamente superior foram eliminados. Por fim, a Figura 8.4(d) apresenta as projeções  $\rho \times z$  e  $\phi \times z$  do evento após a seleção de pontos.

A semelhança com o método de raio de atuação é bastante evidente, uma

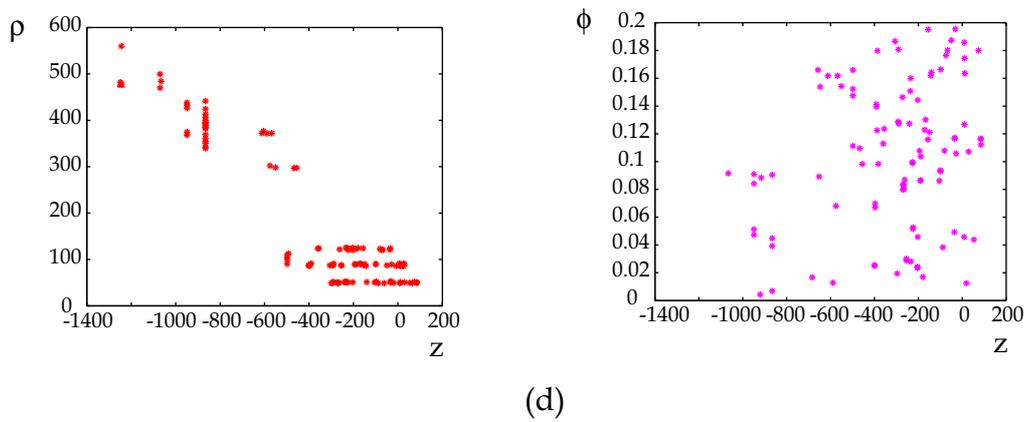
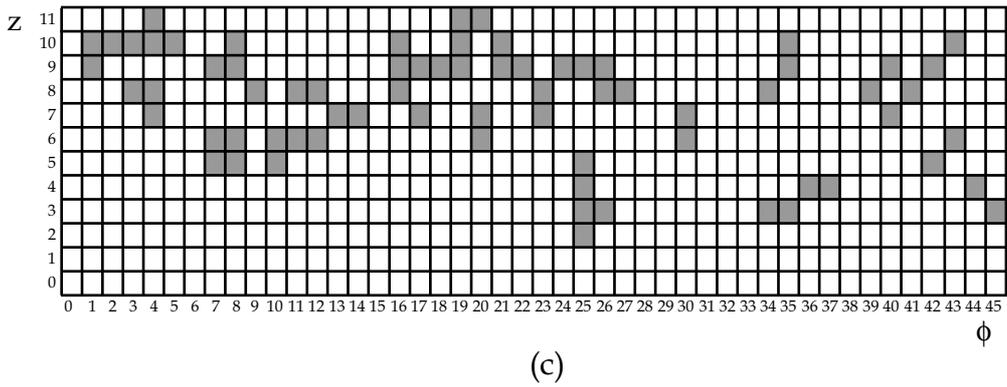
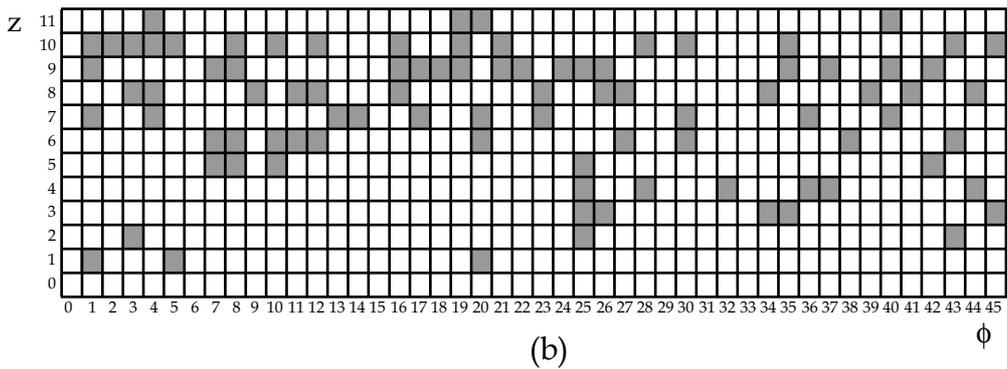
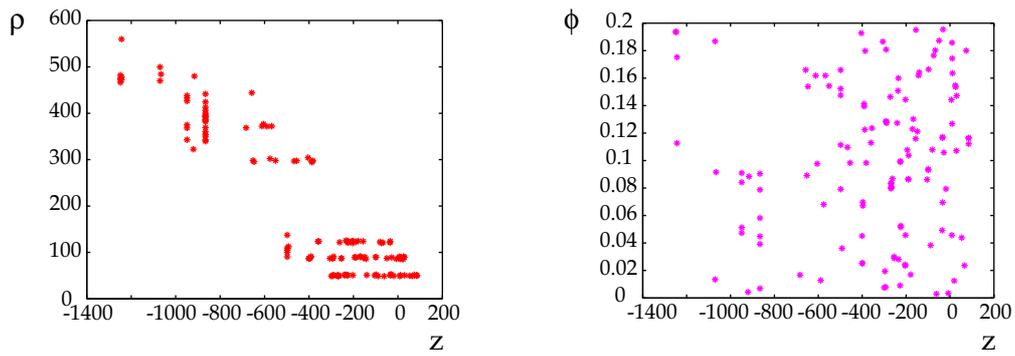


Figura 8.4: Exemplo do método de seleção por vizinhança  $\phi$  aplicado a um evento de alta luminosidade. (a) evento original; (b) mapa bidimensional  $(\phi, z)$ ; (c) pontos selecionados; (d) evento filtrado.

vez que este método também possui um raio de atuação que consiste nos vizinhos imediatos dos pontos. A diferença na seleção de pontos entre os métodos está no erro de quantização, que varia de acordo com a granularidade do espaço  $z \times \phi$ .

## 8.3 Abordagem de Localização

Uma vez selecionados os pontos no evento, a localização do vértice primário é dada por meio da transformada de Hough. Conforme mencionado na introdução deste Capítulo, foram utilizadas três variações desta transformada, são elas: a local, a adaptativa e a morfológica. O Capítulo 4 apresentou os conceitos teóricos relacionados com cada uma destas variantes. Esta Seção apresenta a forma e por que os métodos foram utilizados na solução do problema de localização de vértices primários em regiões de interesse.

Nesta aplicação, o espaço de dados da transformada de Hough é composto pela projeção  $\rho \times z$  do evento, e o espaço de parâmetros é composto pelos coeficientes linear e angular da reta  $z = m\rho + c$ . O vértice primário é determinado pelo coeficiente linear ( $c$ ). Todas as abordagens de localização foram permutadas com todos os métodos de seleção de pontos apresentados na seção 8.2.

### 8.3.1 Transformada de Hough com Conhecimento Especialista

A transformada de Hough pode ou não ser realizada com conhecimento especialista. Quando não há um conhecimento a priori sobre os dados de entrada, estes são combinados dois a dois formando o espaço de Hough. O conhecimento sobre algumas características dos dados de entrada permite a realização da combinação discriminada entre pontos, onde são combinados somente os dados que fazem sentido. Isto significa utilizar um conhecimento especialista na transformada de Hough.

Nessa Tese foram utilizados três métodos especialistas de combinação entre pontos nas transformadas de Hough, que são:  $\rho$  distintos,  $\rho$  vizinhos e do centro da região filtrada para as bordas. Esta Seção aborda com detalhes cada um desses métodos. Todas as variações de transformada de Hough utilizam conhecimento especialista do espaço de dados para a formação do espaço de parâmetros.

O parâmetro  $\rho$  no conjunto de dados de entrada corresponde às camadas dos detectores internos. A combinação entre pontos da mesma camada de detecção introduz ruído no espaço de parâmetros da transformada de Hough, uma vez que pontos muito próximos em  $(\phi, z)$  e pertencentes à mesma camada do detector irão ocasionar coeficientes lineares com valores absolutos muito grandes. Como o coeficiente linear é a resposta do sistema de localização, ou o  $z_{vtx}$  a ser obtido, a ocorrência destes valores dificultam a determinação, ou a localização, feita pelo sistema. Por este motivo, a combinação indiscriminada de pontos não foi considerada no sistema de localização. A Figura 8.5 apresenta um exemplo dos critérios de discriminação de pontos utilizados, que são explanados a seguir.

O critério de combinação de  $\rho$  distintos seleciona os pares de pontos para a transformada de Hough que se encontram em camadas distintas do evento filtrado de entrada. O emprego deste critério é ilustrado na Figura 8.5(a).

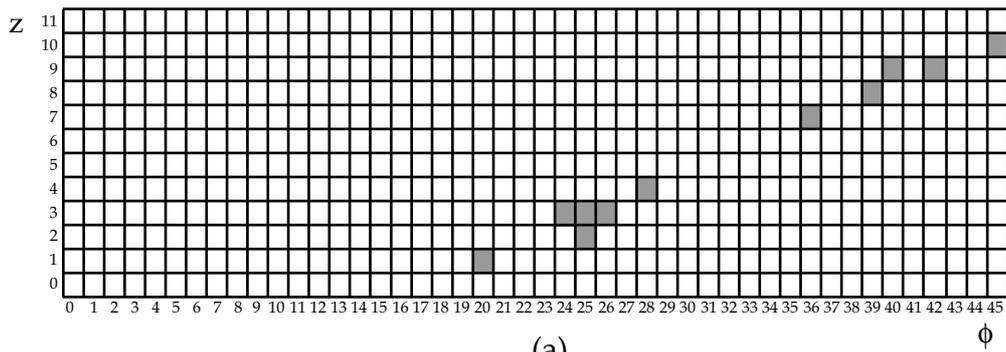
O critério de combinação de  $\rho$  vizinhos seleciona os pares de pontos para a transformada de Hough que se encontram em camadas vizinhas do evento filtrado de entrada. Esta forma de seleção é apresentada na Figura 8.5(b).

Por fim, o critério de combinação do centro da região filtrada para as bordas leva em consideração não somente o parâmetro  $\rho$ , mas também o ponto  $(\phi, z)$  do centro da região filtrada. A combinação dos pontos é dada somente pelo centro para as bordas com pontos de camadas vizinhas ou distintas. A Figura 8.5(c) ilustra este modo de combinação.

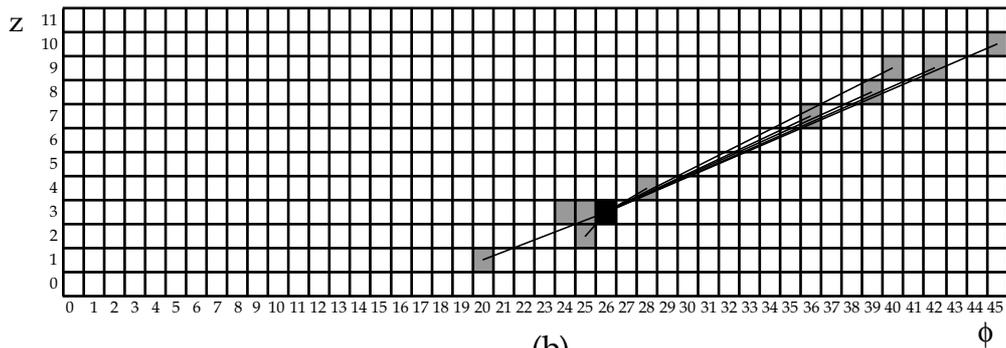
As transformadas de Hough abordadas nessa tese utilizam estes critérios de seleção de pontos. Neste sentido, elas usam uma informação especializada dos dados de entrada. O Capítulo 9 apresenta as diferenças entre as eficiências de localização para sistemas que levam em consideração essas três condições especializadas apresentadas.

### 8.3.2 Transformada de Hough Local - THL

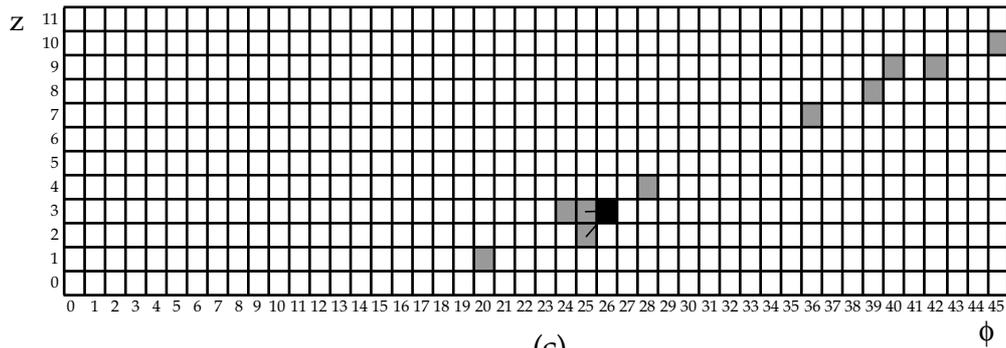
Inicialmente foi testada a transformada de Hough local utilizando os critérios especialistas apresentados na Seção 8.3.1. O espaço bidimensional de parâmetros consiste nos coeficientes angular ( $m$ ) e linear ( $c$ ) da reta definida pela Equação 8.1. A dimensão de  $c$  do espaço de parâmetros varia de evento a evento, de acordo com a pseudorapidez média ( $\mu_\eta$ ). A dimensão de  $m$ , por sua vez, é fixa.



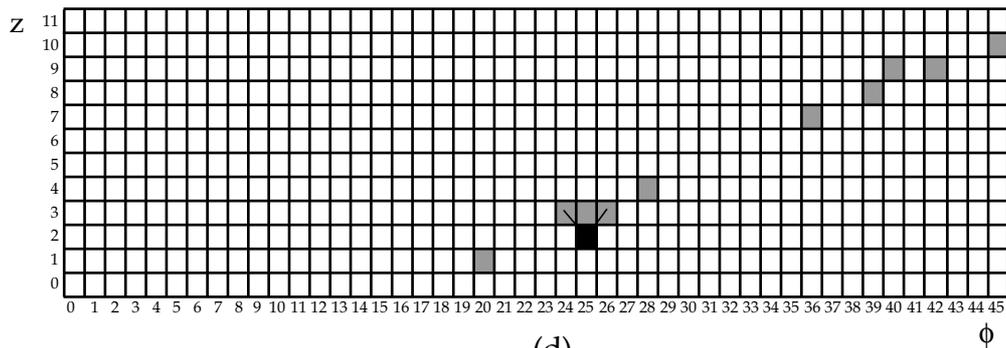
(a)



(b)



(c)



(d)

Figura 8.5: Exemplo de utilização da combinação de pontos na localização do vértice primário. (a) espaço discreto  $z \times \phi$  do evento selecionado pelo método de cone; (b) critério de  $\rho$  distintos; (c) critério de  $\rho$  vizinhos; (d) critério do centro da região selecionada para as bordas.

$$z = m\rho + c \quad (8.1)$$

O diagrama de blocos da Figura 8.6 apresenta o fluxo de execução da THL. As entradas do módulo de localização do vértice primário consistem nos parâmetros de inicialização do sistema e no vetor de pontos, ou SP, filtrados pelo módulo de seleção geométrica. Pode-se observar na Figura 8.6(a) que o módulo de localização possui os blocos de combinação entre pontos e o de THL. O módulo de combinação entre pontos fornece à THL a informação especializada de combinação dos pontos que são utilizados para a montagem do espaço de parâmetros. O diagrama de blocos da THL está representado na Figura 8.6(b). A inicialização da THL fornece os parâmetros para a montagem do histograma, como os valores do intervalo de variação de  $m$  e  $c$  ( $m_{min}, m_{max}, c_{min}, c_{max}$ ) e o número de divisões ou o número de elementos do histograma ( $m_{elementos}, c_{elementos}$ ). Os vetores de SP ( $SP_1(\phi, \rho, z), SP_2(\phi, \rho, z)$ ) representam os pares da transformada de Hough. O espaço de parâmetros é montado por todos os pares de SP e o valor máximo do histograma consiste na saída do sistema,  $z_{vtx}$ .

O espaço de parâmetros da THL varia evento a evento de acordo com os dados de entrada. Nos eventos onde ocorreu erro de localização do vértice foi devido a picos muito próximos no espaço de parâmetros, *i.e.*, o valor correto era vizinho ao encontrado pela THL. Essa característica de erro na localização ocorre devido à precisão do espaço de parâmetros. Um espaço de parâmetros com precisão muito fina proporciona muitas ocorrências com contagens próximas, sendo mais sensível ao ruído e dificultando a localização do vértice. De modo a solucionar o problema de resolução foi testada a transformada de Hough adaptativa.

### 8.3.3 Transformada de Hough Adaptativa - THA

Assim como na THL, a THA também recebe em sua entrada os pares de pontos selecionados por um dos critérios especialistas da Seção 8.3.1. O espaço de parâmetros também é formado pelos coeficientes angulares e lineares da reta da Equação 8.1. O que diferencia a THA da THL é que, na primeira, o intervalo do espaço de parâmetros varia progressivamente de modo a aumentar sua precisão, até atingir uma determinada condição de parada.

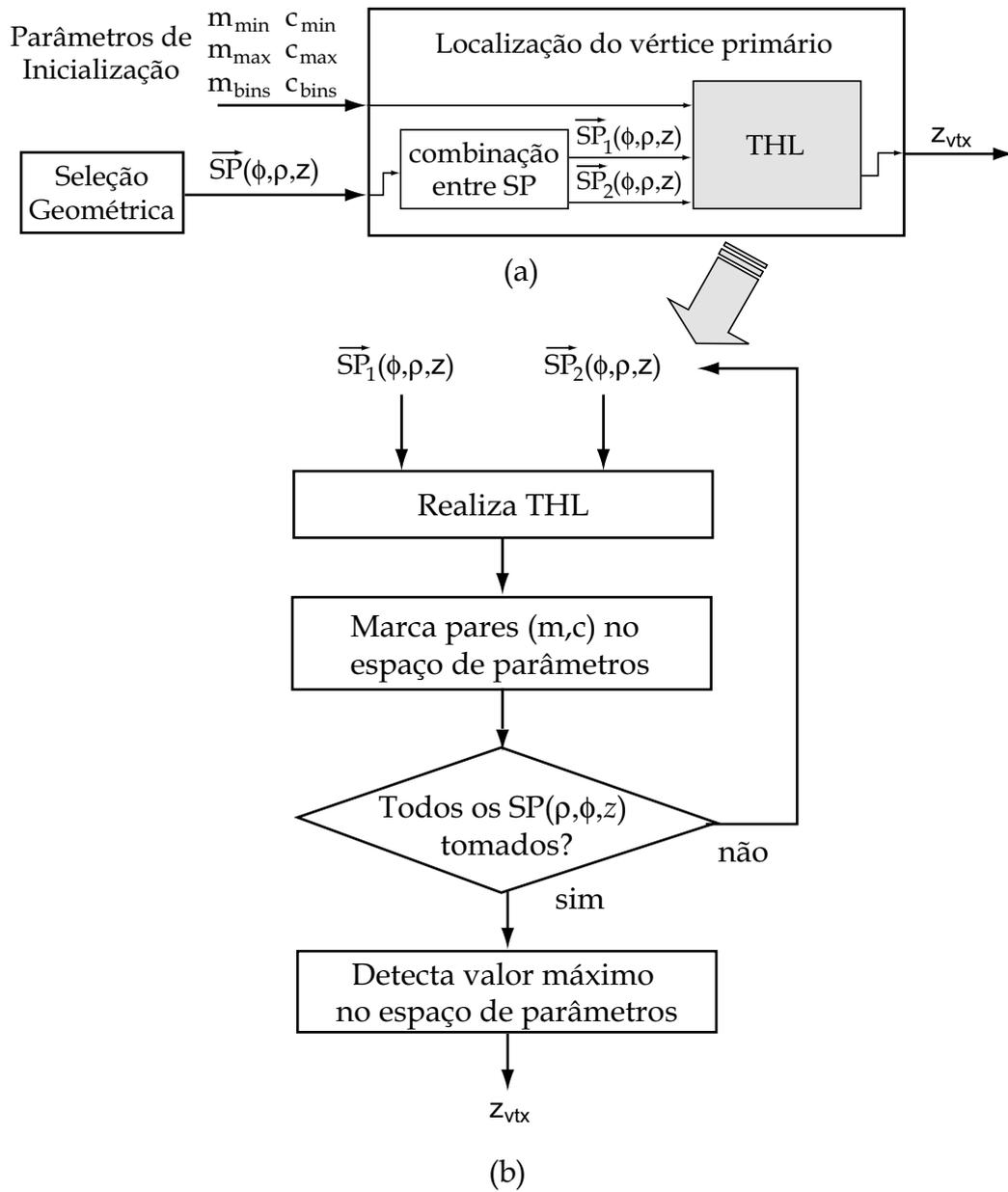


Figura 8.6: Diagrama de blocos da transformada de Hough local.

A condição de parada da THA pode ser fixa ou dinâmica. A condição de parada fixa estabelece que em todos os eventos apenas um número  $x$  de recursões serão realizadas no espaço de parâmetros. A condição de parada dinâmica estabelece critérios de parada para as recursões e, por esta razão, o número de recursões varia evento a evento. Os critérios de parada adotados pela THA são: diferença entre picos maior que um determinado limiar e resolução do espaço de parâmetros menor que um determinado valor.

O critério de parada por diferença entre picos estabelece que a recursão termine quando a diferença entre o pico máximo e o segundo pico do histograma do espaço de parâmetros for maior que um determinado valor ou *threshold*.

A cada recursão os dois primeiros picos são identificados e a diferença entre eles é calculada. Caso este valor da diferença seja superior a um valor fixo estabelecido na inicialização do sistema, o sistema pára e os coeficientes angulares e lineares de saída correspondem ao centro do elemento com o maior pico.

O critério de parada por resolução do espaço de parâmetros exige que a recursão termine quando a resolução do elemento do coeficiente linear seja inferior a um determinado valor ou *threshold*. O valor de parada é determinado na inicialização do sistema e não varia durante a execução.

O diagrama de blocos da Figura 8.7 apresenta o fluxo de execução da THA. Assim como na THL, as entradas do módulo de localização de vértices consistem nos parâmetros de inicialização do sistema e no vetor de pontos, ou SP, filtrados pelo módulo de filtragem geométrica. Podemos observar na Figura 8.7(a) que o módulo de localização possui os blocos de combinação entre pontos e o da THA. O módulo de combinação entre pontos fornece à THA todos os pares de pontos que são utilizados para a montagem do espaço de parâmetros. O diagrama de blocos da THA está representado na Figura 8.7(b). O bloco de Critério de Parada pode ser dividido nos critérios: parada fixa; parada por diferença entre picos; e parada por resolução ou precisão.

A inicialização da THA fornece os parâmetros para: a escolha do critério de parada; a montagem do histograma e; o número de recursões máximo permitido. Os vetores de SP ( $SP_1(\phi, \rho, z), SP_2(\phi, \rho, z)$ ) representam os pares da transformada de Hough. São calculados todos os valores de  $m$  e  $c$  a serem inseridos no espaço

de parâmetros, em seguida o processo de recursão da THA inicia. É montado o histograma do espaço de parâmetros, em seguida é detectado o pico do histograma e o critério de parada é verificado. Caso tenha sido atingida a condição de parada o valor do coeficiente angular ( $c$ ) do máximo do histograma corresponde a saída  $z_{vtx}$ .

A transformada de Hough adaptativa acertou 73,4% dos eventos acertados pela THL, como apresentado no diagrama de Venn da Figura 8.8. A THA acertou diversos eventos onde a THL não obteve sucesso, devido ao ajuste do espaço de parâmetros. Os eventos onde a THL acertou e a THA errou foram aqueles em que o pico máximo da primeira recursão da THA não representava a reta verdadeira. A THL acertou nesses casos, pois seu espaço de parâmetros possuía maior resolução que a primeira recursão da THA. Desta forma, o pico máximo representava o valor correto dos coeficientes da reta desejada.

O estudo de casos dos erros da THA apontou que estes ocorriam sistematicamente na primeira recursão, onde ocorriam picos no espaço de parâmetros provenientes do ruído dos dados. Foi observado que o pico verdadeiro encontrava-se sempre na vizinhança do de valor máximo. No entanto, ele nem sempre era o primeiro vizinho, mas encontrava-se na maioria dos casos nas proximidades do pico máximo. De forma a buscar solucionar esse problema da THA foi testada a transformada de Hough morfológica (THM). Na THM uma máscara é utilizada no pico máximo do espaço de parâmetros, de modo que todos os elementos marcados como 1 na máscara são considerados na recursão seguinte da transformada. A próxima seção apresenta os detalhes de implementação da THM e algumas considerações sobre seu funcionamento.

### 8.3.4 Transformada de Hough Morfológica - THM

Assim como a THL e a THA, a transformada de Hough morfológica recebe em sua entrada os pares de pontos selecionados por um dos critérios especialistas da Seção 8.3.1 e o espaço de parâmetros é formado pelos coeficientes angulares e lineares da reta da Equação 8.1. Contudo, a THM aplica uma máscara no espaço de parâmetros antes da variação responsável pelo aumento de precisão. O objetivo desta máscara é levar em consideração informações da vizinhança do pico máximo do histograma. A THL e THA consideram apenas o valor de máximo, desconsiderando

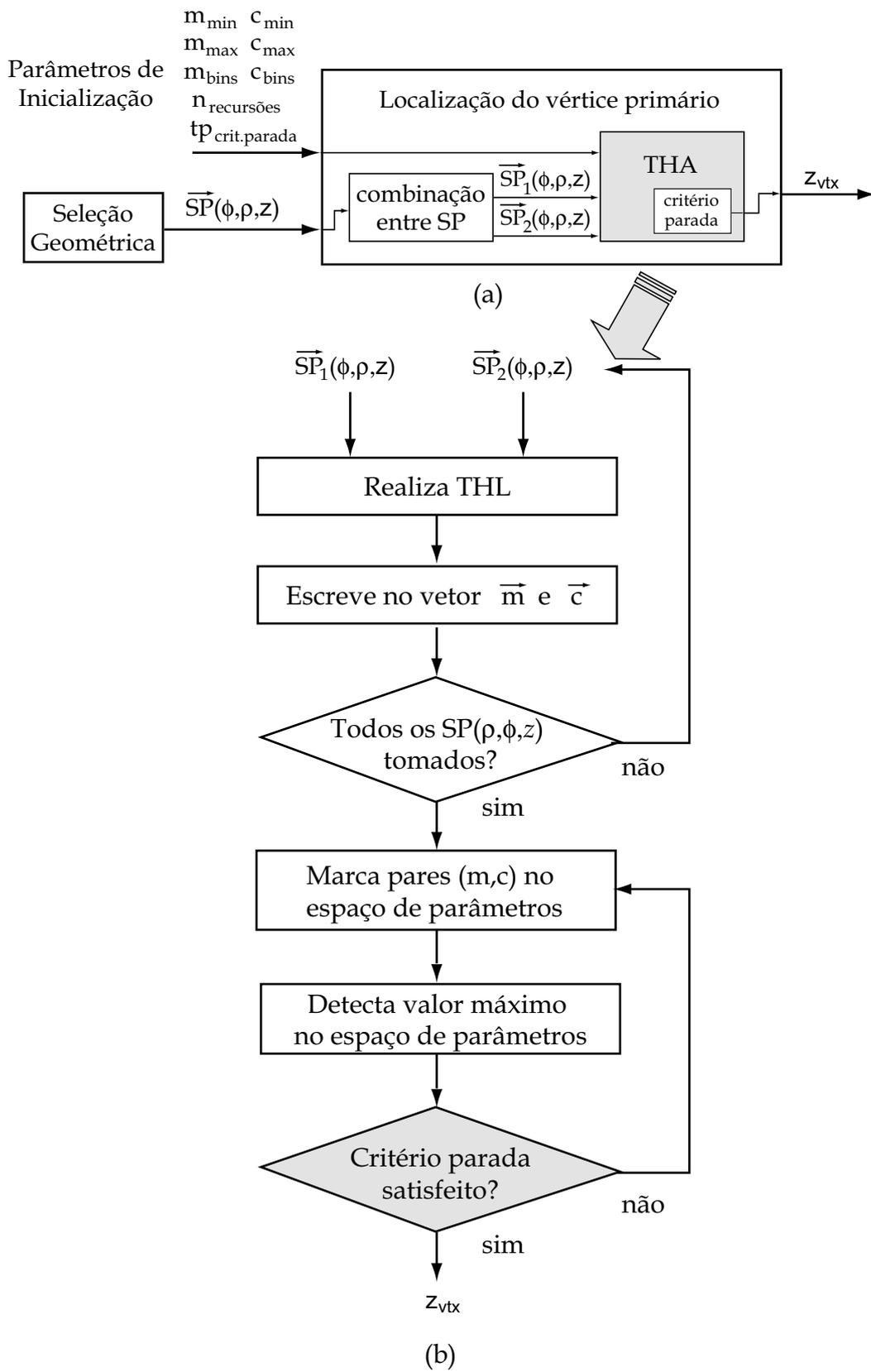


Figura 8.7: Diagrama de blocos da transformada de Hough adaptativa.

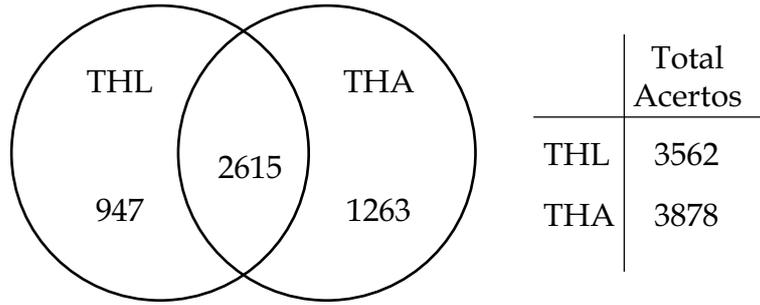


Figura 8.8: Diagrama de Venn das transformadas de Hough local e adaptativa.

qualquer influência dos valores vizinhos muito próximos. Em geral, nestas situações, os elementos muito próximos representam o mesmo ponto, mas que em função de um ruído foram desmembrados.

Foram utilizados na THM os mesmos critérios de parada da THA, tendo em vista verificar a eficiência da THM nos casos de erro da THA. Diversas máscaras foram testadas no conjunto de treino dos eventos. Foi verificado que aquelas que consideravam dois ou mais vizinhos na horizontal e um ou dois vizinhos na vertical ao pico de valor máximo obtiveram as maiores eficiências. Desta forma, as máscaras apresentadas na Figura 8.9 representam as máscaras utilizadas no espaço de parâmetros.

O diagrama de blocos da Figura 8.10 apresenta o fluxo de execução da THM. Assim como nos casos anteriores, as entradas do módulo de localização de vértices consistem nos parâmetros de inicialização do sistema e no vetor de pontos, ou SP, filtrados pelo módulo de filtragem geométrica. Podemos observar na Figura 8.10(a) que o módulo de localização possui os blocos de combinação entre pontos e o da THM.

O diagrama de blocos da THM está representado na Figura 8.10(b). A inicialização da THM fornece os parâmetros para: a escolha do critério de parada; a montagem do histograma; o número de recursões máximo permitido e; a máscara morfológica  $\mathbf{M}$  a ser aplicada no espaço de parâmetros. Os vetores SP, representados por  $(SP_1(\phi, \rho, z), SP_2(\phi, \rho, z))$ , formam os pares da transformada de Hough. São calculados todos os valores de  $m$  e  $c$  a serem inseridos no espaço de parâmetros, em seguida o processo de recursão da THM inicia. É montado o histograma do espaço de parâmetros, em seguida é detectado o valor máximo do histograma, o centro da máscara  $\mathbf{M}$  é posicionado no pico máximo do histograma, os valores da máscara são

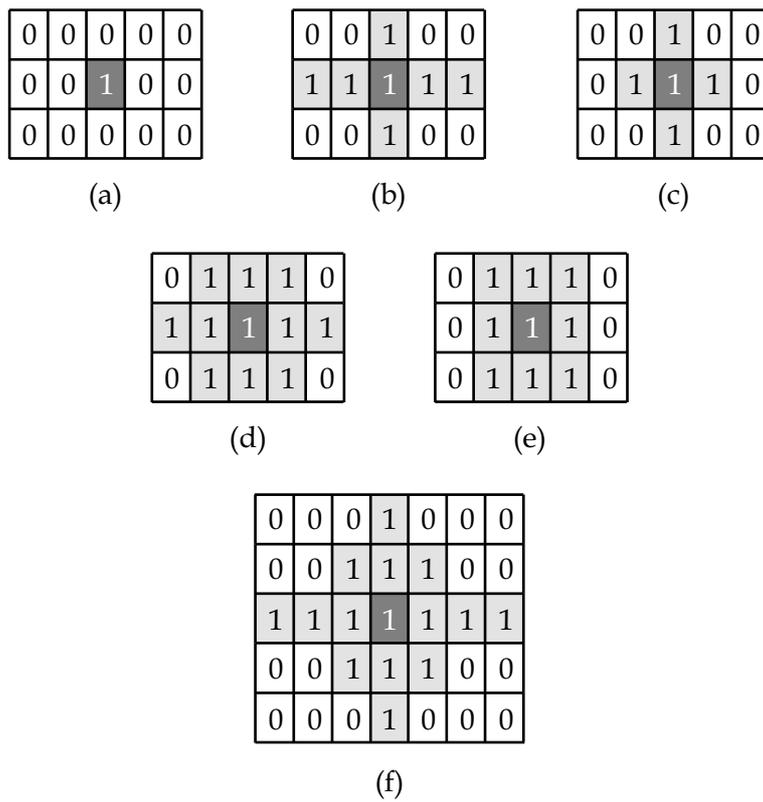


Figura 8.9: Ilustração das máscaras utilizadas na THM. (a) até (e) ilustram as máscaras utilizadas nos espaços de parâmetro  $5 \times 10$ ; (f) máscara utilizada no espaço de parâmetros  $20 \times 100$ .

multiplicados com os do histograma e o critério de parada é verificado. Caso tenha sido atingida a condição de parada, o algoritmo fornece a saída  $z_{vtx}$ .

A THM acertou todos os eventos corretos da THA e 78,9% dos eventos da THL. A Figura 8.11 apresenta o diagrama de Venn das três transformadas. Conforme esperado, a THM solucionou os casos em que o elemento da reta correta encontrava-se na vizinhança do pico máximo do histograma.

Um estudo de casos dos erros obtidos na THM revelou que esta transformada é sensível aos eventos com múltiplos pontos na camada mais externa do detector. Esta sensibilidade ocorre com todas as outras variações da transformada de Hough. A ocorrência de múltiplos pontos na camada mais externa gera um histograma com picos muito próximos, sendo bastante sensível à determinação daquele que representa a reta verdadeira.

## 8.4 Implementação dos Métodos

A aplicação dos métodos de seleção e localização apresentados nas Seções 8.2 e 8.3, foi utilizada em eventos de elétrons individuais com empilhamento. Ao todo dez mil eventos foram testados para combinações diversas de métodos de seleção e sistemas de localização.

O diagrama de blocos da Figura 8.12 apresenta o esquema do sistema de reconstrução de vértices primários. Este sistema é composto por três módulos, que são: leitura dos dados de entrada; filtragem geométrica e; localização do vértice primário. O módulo de leitura dos dados de entrada recebe como parâmetros de inicialização o tipo de evento a ser lido, *i.e.*, arquivos com eventos de elétrons individuais à alta luminosidade ou à baixa luminosidade, e quantos eventos serão lidos. A saída deste módulo consiste em vetores com os SP dos eventos.

O módulo de seleção geométrica dos SP recebe como entrada a saída do módulo de leitura e os parâmetros de inicialização. Os parâmetros consistem: na escolha do tipo de filtro; no raio  $\phi$ ; no vértice do cone  $e$ ; a pseudorapidez média  $\mu_\eta$ . Dependendo da escolha do filtro alguns parâmetros de entrada não são utilizados. A saída deste módulo consiste no vetor de pontos SP que permaneceram após a aplicação do filtro.

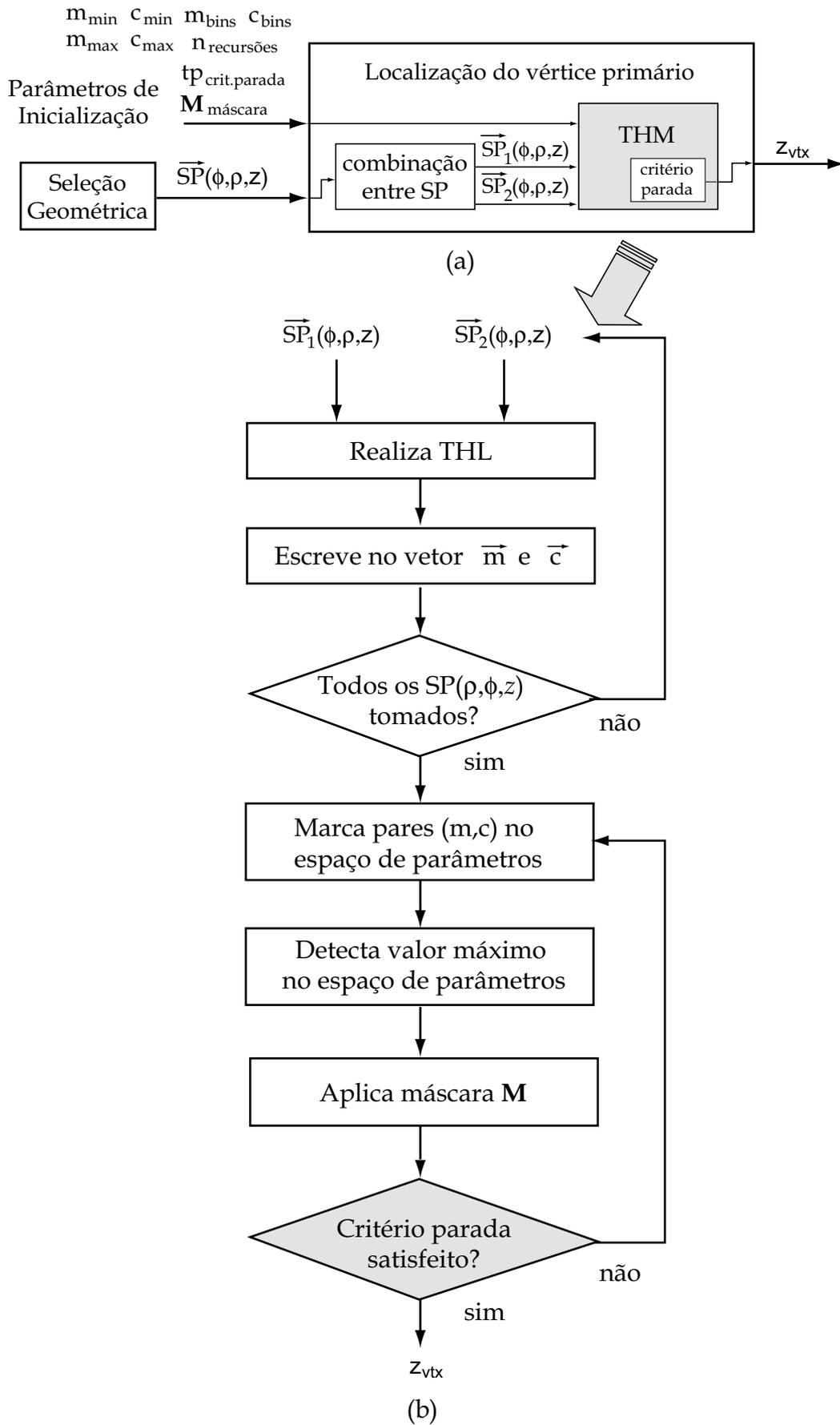


Figura 8.10: Diagrama de blocos da transformada de Hough morfológica.

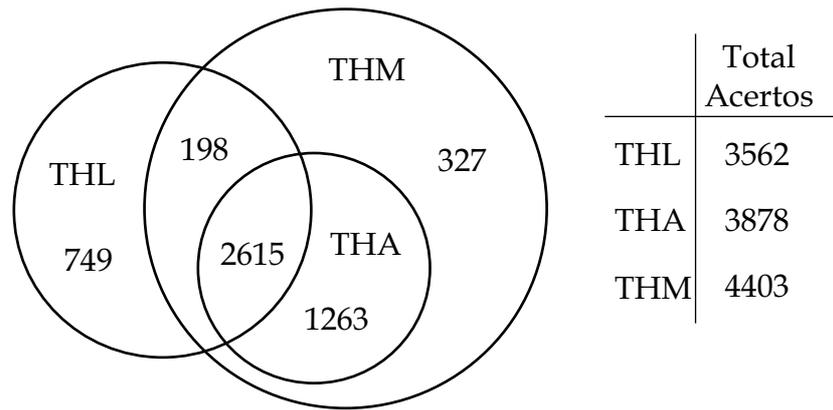


Figura 8.11: Diagrama de Venn das três transformadas.

O módulo de localização de vértice primário é composto por dois grandes blocos, que são: escolha de pontos e localização. O bloco de escolha de pontos é constituído pelos blocos:  $\rho$  distintos;  $\rho$  vizinhos e; critério do centro para as bordas. O bloco de localização é formado pelos blocos de transformada de Hough (THL, THA e THM) e pelo bloco de condição de parada. O bloco de condição de parada é utilizado somente nas abordagens adaptativa e morfológica da transformada de Hough e é subdividido em: fixo; diferença entre picos e resolução ou precisão do espaço.

Os parâmetros de entrada do sistema de localização configuram todos os blocos, determinando o método de escolha de pontos, a transformada de Hough e a condição de parada. Além destes parâmetros, a inicialização fornece todos os parâmetros de entrada de cada um dos blocos possíveis de configuração do sistema.

O Capítulo 9 apresentará as eficiências das combinações deste sistema, os gráficos de tempo de processamento total e de cada bloco, ambos escalados para uma máquina de 4GHz.

## 8.5 Reconstrução de Vértices Primários no Ambiente ATHENA

O ambiente de simulação do ATLAS, o ATHENA (veja Seção 2.4.4.1), encontra-se em processo de contínuo desenvolvimento. Neste ambiente são testados todos os algoritmos do sistema de filtragem do ATLAS (HLT) e toda a seqüência de execução

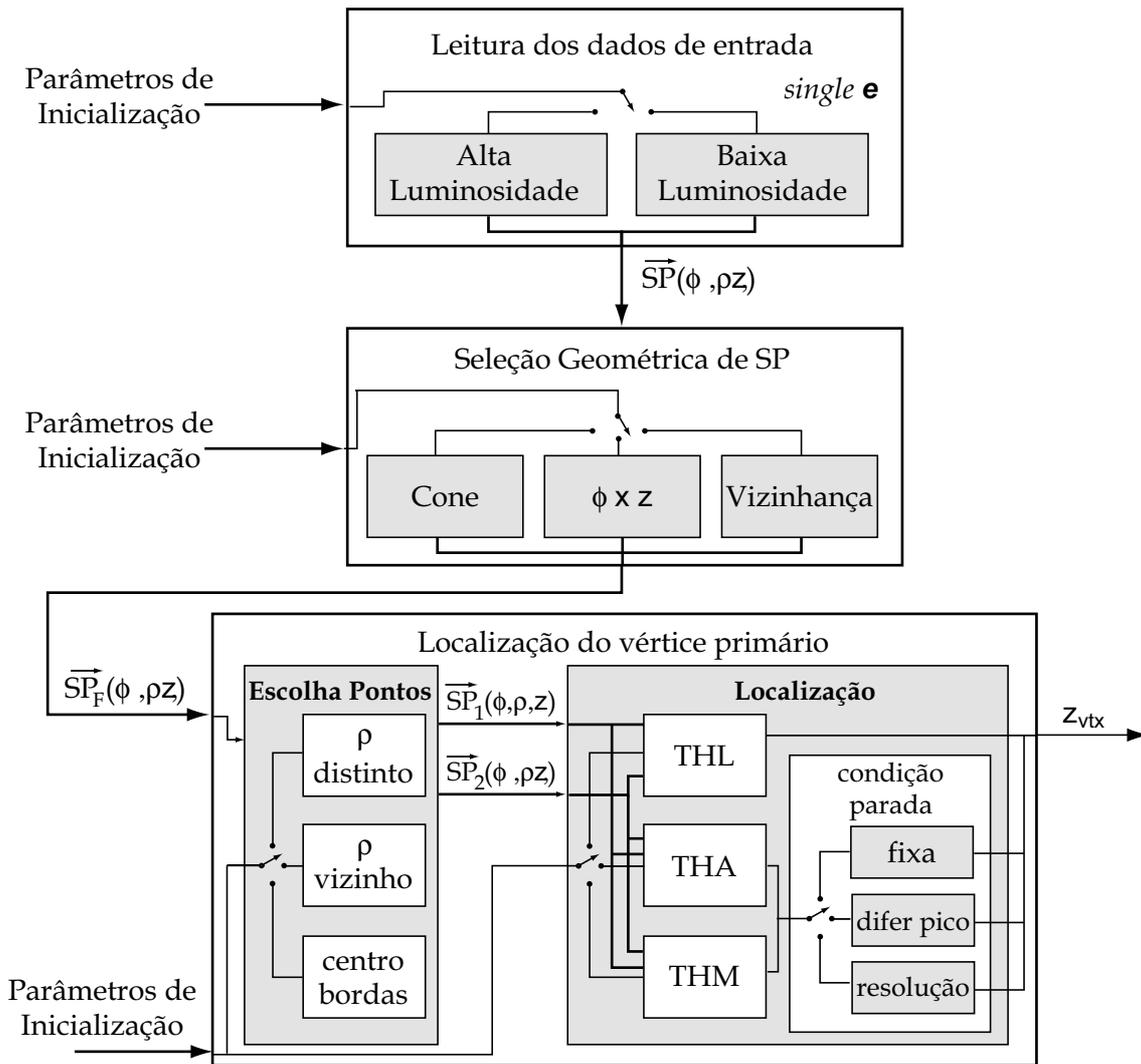


Figura 8.12: Diagrama de blocos do sistema de reconstrução de vértices primários.

do experimento é simulada.

De modo a testar o sistema de reconstrução de vértices desenvolvido foram realizadas as devidas substituições das rotinas do ZFinder por aquelas implementadas nessa tese. Cada possibilidade de permutação dos blocos do sistema da Figura 8.12 foi incorporada ao ZFinder por meio de funções.

O conjunto de dados de entrada representa elétrons simples com empilhamento a 25GeV de luminosidade. É importante observar que este conjunto é diferente daquele utilizado no sistema da Figura 8.12. No sistema da Figura 8.12, os dados de entrada, sejam eles de alta ou baixa luminosidade, correspondem à resposta da simulação do primeiro estágio de filtragem. Na situação relacionada com a execução no ATHENA, tem-se que são gerados novos eventos simulados pelo método Monte Carlo. Neste sentido, o resultado da eficiência do sistema no ATHENA pode apresentar uma pequena diferença quando comparado com aquele obtido no sistema *stand-alone*, ou da Figura 8.12.

Assim como na versão *stand-alone*, o ATHENA possui uma rotina de medida de tempo de execução. Todas as variações do sistema de localização por transformada de Hough foram testadas no ambiente ATHENA. Além do tempo de execução, foi verificada a eficiência de localização do vértice primário, que é o foco do desenvolvimento das rotinas de localização. Esta eficiência foi comparada com a do ZFinder.

O Capítulo 9 apresenta os tempos médios de execução medidos no ATHENA, assim como a eficiência dos sistemas para este conjunto de dados de elétrons individuais com empilhamento à 25GeV.

## 8.6 Reconstrução de Vértices Primários no SHARC ADSP-21160

Os DSPs são microprocessadores de sinais digitais com características próprias que podem ser programados e operam em tempo real com velocidades muito superiores aos microprocessadores para aplicações genéricas. Eles otimizam as operações de multiplicação e acumulação, envolvidas no produto interno, realizando-as, de preferência, em um ciclo. Esta otimização se faz através de estruturas de endereçamento

flexível e de unidades lógico-aritméticas de alto desempenho.

Além de contarem com um barramento duplo, o de dados e o de programa, e apresentarem unidades internas que trabalham paralelamente, os DSPs contam com uma razoável quantidade de memória interna permitindo que toda a aplicação esteja em seu interior. Estes microprocessadores possuem funções comuns aos processadores de uso geral com exceção das mais complexas: trigonométricas, exponenciais e logarítmicas que são normalmente constituídas por séries numéricas ou por tabelas (Look-Up-Tables).

Os DSPs são comumente utilizados em: processamento de imagens, redes neurais, compressão de dados, filtragem, reconhecimento de voz, etc.

Conforme apresentado no Capítulo 2, o LVL1 corresponde ao primeiro estágio de filtragem do ATLAS, cuja implementação é feita em *hardware*. Por outro lado, o LVL2 é implementado em *software* e precisa ser executado em tempo real. Sob esta ótica, qualquer contribuição de implementação dos programas do LVL2 em *hardware* tende a ser bem recebida.

Uma vez que o sistema de localização de vértices primários corresponde ao primeiro estágio de execução do LVL2 do ATLAS, exatamente na transição dos níveis LVL1 e LVL2, entende-se que é interessante conhecer o desempenho do sistema de localização quando implementado em *hardware*. Assim, o algoritmo que obteve melhor eficiência e tempo de execução foi selecionado para ser testado no kit de desenvolvimento SHARC ADSP-21160 EZ-KIT.

A placa de desenvolvimento possui dois processadores SHARC, o ADSP-21160M e o ADSP-21160N, uma interface USB, memória SBSRAM com 512 Kb (64 K x 32 bits x 2-chips), memória flash de 512 Kb (512 K x 8 bits), conectores de interface e portas de entrada e saída. O ADSP-21160M possui 80 MHz de velocidade e o ADSP-21160N possui 95 MHz de velocidade.

Devido à capacidade de armazenamento restrita do kit de desenvolvimento, não foi possível testar o sistema de localização de forma completa. Somente o módulo de localização foi testado. Desta forma, os dados de entrada enviados ao DSP consistem nos vetores  $\mathbf{m}$  e  $\mathbf{c}$ .

O DSP realizou os passos de montagem do histograma do espaço de parâmetros e determinação do  $z_{vtx}$ . O algoritmo escolhido para ser utilizado no DSP teve seu

código adaptado de modo a tirar proveito dos recursos oferecidos por este tipo de processador. No próximo Capítulo serão apresentados: os resultados obtidos no kit DSP, evidenciando as propriedades de tempo de processamento e de eficiência de localização.

# Capítulo 9

## Apresentação, Análise e Interpretação dos Resultados

Este capítulo aborda os resultados das análises apresentadas no Capítulo 8. Os resultados foram obtidos utilizando dados simulados de elétrons individuais com empilhamento e alta luminosidade. Como esse sistema de reconstrução de vértices primários é executado em tempo real no segundo nível de filtragem (LVL2) do ATLAS, os algoritmos devem obedecer à restrição de tempo de execução estabelecida pela colaboração. Esta restrição consiste no tempo máximo de 1 ms em uma máquina de dois processadores de 4 GHz. Além da restrição no tempo de execução, a eficiência da reconstrução deve ser maior ou igual a do algoritmo de referência (*ZFinder*).

Os resultados deste capítulo apresentam medidas de tempo de execução e eficiência de localização do vértice. Cada bloco do sistema de reconstrução, representado na Figura 8.12, foi testado de forma independente. O tempo de execução dos algoritmos foi escalado para uma máquina de dois processadores de 4GHz. O cálculo da eficiência é realizado subtraindo o vértice obtido com o vértice alvo obtido da simulação, ou valor desejado. Um acerto é considerado quando a distância entre o vértice desejado e o obtido for menor que  $1mm$ .

Foram testadas todas as combinações dos módulos de seleção de amostras e localização de vértices, representados na Figura 8.12. A figura de mérito consistiu em se obter uma eficiência superior a do algoritmo referência (*ZFinder*) com um tempo de execução igual ou inferior ao dele.

As próximas seções apresentam as análises realizadas nos sistemas de seleção de amostras e localização do vértice primário de colisão, representados na Figura 8.12. Inicialmente são apresentadas tabelas com os tempos de execução e eficiência de todas as combinações. Em seguida, são apresentados gráficos detalhando o funcionamento da configuração que obteve a melhor eficiência.

## **9.1 Reconstrução de Vértices Primários para Elétrons com Empilhamento**

A reconstrução de vértices primários para elétrons com empilhamento utiliza eventos de elétrons individuais com empilhamento e alta luminosidade. O conjunto de dados utilizado foi dividido em desenvolvimento e validação. Para tal, um total de dez mil eventos foi dividido igualmente em duas partes. As amostras do conjunto de desenvolvimento foram utilizadas para ajustar os parâmetros dos módulos de seleção de amostras e de localização de vértices primários. Os eventos de validação, por sua vez, foram utilizados para verificar a eficiência do sistema. Os resultados apresentados referem-se aos eventos de validação.

Foram medidos os tempos de processamento dos módulos de seleção de amostras para a função de localização e ambos para os eventos de validação. Os resultados representados nas Tabelas 9.2, 9.3 e 9.4 ilustram os tempos de processamento e eficiências para todas as combinações entre blocos do diagrama da Figura 8.12.

### **9.1.1 Eficiências e Tempos de Processamento**

Podemos observar nas Tabelas 9.2, 9.3 e 9.4, que a transformada de Hough morfológica obteve as melhores eficiências, quando comparada com as versões adaptativa e local. A transformada morfológica explora a característica tridimensional do espaço de parâmetros, acertando eventos que são perdidos pelas outras versões de transformadas.

Observamos também que o método de seleção de pontos por cone obteve os menores tempos de execução. No entanto, as configurações que utilizam este método de seleção não possuem eficiências superiores a 52,50%. A queda na eficiência ocorre, pois este método de cone restringe em muito o universo de pontos dos eventos. O

espaço de parâmetros da transformada de Hough fica com menos ruído de fundo; no entanto, o nível do sinal torna-se equivalente ao do ruído, fazendo com que o método seja mais sensível.

O método de seleção por raio de atuação em  $\phi \times z$  (Tabela 9.3) possui em média, latência de 3,06 ms para eventos de alta luminosidade. Houve um aumento na eficiência de  $\approx 20\%$  com relação aos sistemas que utilizaram a seleção por cone. Isto se deve ao fato deste método selecionar somente os pontos que estão próximos em um determinado raio de atuação, não reduzindo de forma tão significativa os pontos nos eventos. Esta pequena redução no número de amostras faz com que haja mais pares de pontos no espaço de dados da transformada de Hough, aumentando o nível de sinal no espaço de parâmetros. O nível de ruído não aumenta na mesma proporção que o de sinal devido aos métodos de escolha de pontos ( $\rho$  distinto (RD);  $\rho$  vizinho (RV); e centro bordas (CB)).

É importante ressaltar que os métodos de escolha de pontos determinam regras para combinação de pontos no espaço de dados da transformada de Hough, *i.e.*, a combinação dos pontos não é realizada de forma indiscriminada. Pontos da mesma camada do detector nunca são combinados em nenhum dos métodos de escolha de pontos. As diferenças quanto à aplicação dos métodos de escolha de pontos são verificadas nas Tabelas 9.2, 9.3 e 9.4 pelas iniciais RD, RV e CB. A Figura 9.1 ilustra a média e o desvio padrão das eficiências de localização para cada método de escolha de pontos (RD, RV e CB) de acordo com os métodos de seleção de pontos (cone (CN); raio de atuação  $\phi \times z$  (FZ); vizinhança (VZ)). Nesta Figura, o eixo da abscissa consiste nos três tipos de sistema de seleção de pontos (CN, FZ e VZ). O eixo da ordenada representa a porcentagem de acerto do módulo de localização onde a legenda mostra a eficiência com relação aos diferentes módulos de escolha de pontos (RD, RV e CB).

O gráfico da Figura 9.1 ilustra que não há diferença significativa na eficiência quando são aplicados os diferentes métodos de escolha de pontos. Desta forma, os três métodos de escolha de pontos podem ser considerados equivalentes, uma vez que eles não alteram de forma significativa a eficiência do sistema de localização. A combinação indiscriminada de pontos está representada pelo símbolo Indiscriminado no gráfico desta Figura. A combinação indiscriminada de pontos eleva o ruído no

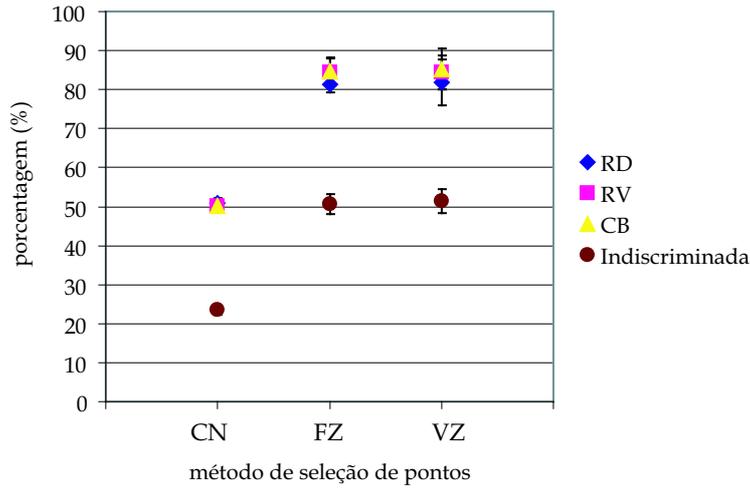


Figura 9.1: Eficiência média e RMS dos métodos de localização de vértices (RD, RV e CB) de acordo com os métodos de seleção de pontos (CN, FZ, VZ).

espaço de parâmetros de Hough, ocasionando uma redução na eficiência de localização.

A Figura 9.2 ilustra os tempos de processamento dos métodos em milissegundos para os mesmos eventos de alta luminosidade representados na Figura 9.1. Os tempos de execução são equivalentes para os sistemas de localização de vértices com os diferentes sistemas de escolha de pontos (RD, RV e CB). As variações significativas observadas nos tempos ocorrem para os sistemas de seleção de pontos, onde o cone (CN) possui tempos em média 3 vezes menor que os métodos de raio de atuação em  $\phi \times z$  e vizinhança.

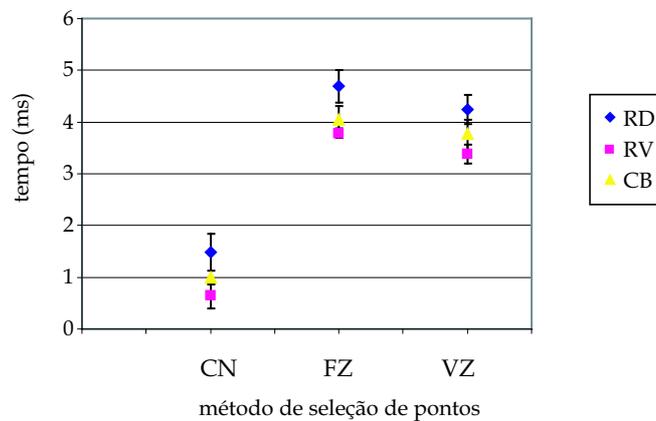


Figura 9.2: Tempo médio de execução dos métodos de localização de vértices (RD, RV e CB) de acordo com os métodos de seleção de pontos (CN, FZ, VZ).

O método de seleção por raio de atuação em  $\phi \times z$  (FZ) possui eficiências que variam entre 76 e 86%, desconsiderando os valores obtidos pela combinação indiscriminada de pontos. A Tabela 9.3 apresenta os tempos de execução e as eficiências de localização para todas as combinações representadas no diagrama da Figura 8.12. Conforme apresentado nas Figuras 9.1 e 9.2, tanto as eficiências de localização quanto os tempos de processamento para os diferentes métodos de escolha de pontos são equivalentes.

O método de seleção de pontos por vizinhança (VZ), apresentado na Tabela 9.4 possui eficiências que variam entre 73 e 89%. O tempo de processamento possui uma média de  $\approx 4 (\pm 3)$  ms. O método de vizinhança aliado com: a escolha de pontos Centro Borda (CB); a transformada de Hough morfológica (THM); e com o número de recursões fixo no espaço de parâmetros foi o método de obteve maior eficiência. Quando comparado com o algoritmo referência *ZFinder*, este método possui tempo de execução menor que a metade do tempo do *ZFinder* ( $\approx 10 (\pm 2)$  ms) e eficiência ligeiramente superior (4356/4942 - 88,14%).

Através da análise das Tabelas 9.2, 9.3 e 9.4, é possível constatar que todos os métodos desenvolvidos possuem tempos de execução menores que a metade do tempo do algoritmo referência. É importante ressaltar que cada medida de tempo apresentada nas Tabelas 9.2, 9.3 e 9.4, correspondem a uma média de 100 execuções do mesmo evento. Desta forma, foi reduzido o ruído de medida de tempo do programa, uma vez que não há como garantir que um único processo esteja sendo executado na máquina. De modo se obter melhor compreensão das Tabelas 9.2, 9.3 e 9.4, foi criada a Tabela 9.1 com a legenda das siglas adotadas. O método de seleção por cone apresenta os menores tempos de execução. O raio utilizado neste método foi mantido fixo em 1 mm e a posição do vértice foi aproximada através de cálculos realizados com os parâmetros de simulação. É importante ressaltar que a aplicação do vértice do calorímetro e um ajuste fino do raio do cone podem aumentar o valor da eficiência de localização.

Sigla	Significado
CN	Seleção de pontos por cone
FZ	Seleção de pontos por raio $\phi \times z$
VZ	Seleção de pontos por vizinhança em $\phi$
RD	Escolha de pontos $\rho$ distinto
RV	Escolha de pontos $\rho$ vizinho
CB	Escolha de pontos centro bordas
THL	Transformada de Hough local
THA	Transformada de Hough adaptativa
THM	Transformada de Hough morfológica
FX	Condição de parada fixa
DP	Condição de parada por diferença entre picos
RS	Condição de parada por resolução

Tabela 9.1: Legenda das siglas utilizadas nas Tabelas 9.2, 9.3 e 9.4.

### 9.1.2 Comparação entre os Métodos de Reconstrução por Hough e o *ZFinder*

Na seção 9.1 foram apresentadas e analisadas todas as variações dos sistemas de seleção geométrica de pontos e localização de vértices primários para eventos de elétrons individuais com empilhamento e a alta luminosidade. Foi verificado que a configuração de seleção de pontos por vizinhança aliada com a transformada de Hough morfológica possui a melhor eficiência de localização (VZCBTHMFX, de acordo com a Tabela 9.4).

O objetivo desta seção consiste em apresentar os gráficos de comparação entre os métodos de seleção geométrica de pontos e do método de sistema de localização aplicados em eventos de elétrons individuais com empilhamento com alta ou baixa luminosidade. Da mesma forma, serão apresentados gráficos dos tempos de execução das transformadas de Hough.

A Figura 9.3 apresenta os histogramas de tempo de processamento dos métodos de seleção geométrica de pontos. A coluna  $t_{SP}(ms)$  das Tabelas 9.2, 9.3 e 9.4 na coluna  $t_{SP}(ms)$  correspondem aos valores médios e RMS destes histogramas para cada configuração do sistema de reconhecimento da Figura 8.12.

Configuração			$t_{SP}$ (ms)	$t_{LV}$ (ms)	$t_{total}$ (ms)	Eficiência	
CN	RD	THL		0,0392(0,0131)	0,0490(0,0163)	0,1190(0,0761)	2595/4942 (52,50%)
		THA	FX	0,0382(0,0115)	0,0530(0,0289)	0,0747(0,0524)	2458/4942 (49,72%)
			DP	0,0379(0,0115)	0,0509(0,0226)	0,0681(0,0651)	2457/4942 (49,70%)
			RS	0,0405(0,0153)	0,0498(0,0234)	0,0707(0,0564)	2457/4942 (49,70%)
		THM	FX	0,0380(0,0115)	0,0571(0,0344)	0,1505(0,1149)	2550/4942 (51,59%)
			DP	0,0379(0,0115)	0,0548(0,0330)	0,1379(0,1116)	2549/4942 (51,56%)
	RS		0,0378(0,0114)	0,0545(0,0327)	0,1325(0,1126)	2549/4942 (51,56%)	
	RV	THL		0,0379(0,0115)	0,0157(0,0111)	0,0431(0,0960)	2501/4942 (50,59%)
		THA	FX	0,0379(0,0115)	0,0123(0,0076)	0,0339(0,1014)	2439/4942 (49,35%)
			DP	0,0377(0,0114)	0,0107(0,0064)	0,0304(0,0847)	2438/4942 (49,32%)
			RS	0,0377(0,0114)	0,0109(0,0070)	0,0316(0,0962)	2438/4942 (49,32%)
		THM	FX	0,0379(0,0114)	0,0278(0,0118)	0,0716(0,0906)	2516/4942 (50,91%)
			DP	0,0377(0,0114)	0,0246(0,0106)	0,0690(0,1290)	2515/4942 (50,88%)
	RS		0,0377(0,0114)	0,0252(0,0126)	0,0710(0,1077)	2515/4942 (50,88%)	
	CB	THL		0,0379(0,0115)	0,0344(0,0220)	0,0961(0,0718)	2502/4942 (50,62%)
		THA	FX	0,0377(0,0114)	0,0255(0,0225)	0,0782(0,0650)	2441/4942 (49,38%)
			DP	0,0377(0,0114)	0,0220(0,0197)	0,0701(0,0704)	2439/4942 (49,34%)
			RS	0,0378(0,0114)	0,0214(0,0208)	0,0703(0,0685)	2439/4942 (49,34%)
		THM	FX	0,0376(0,0114)	0,0266(0,0219)	0,0860(0,0687)	2518/4942 (50,94%)
			DP	0,0394(0,0117)	0,0215(0,0189)	0,0704(0,0615)	2515/4942 (50,89%)
RS	0,0379(0,0114)		0,0181(0,0169)	0,0702(0,0624)	2515/4942 (50,89%)		

Tabela 9.2: Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção de Cone. Os valores entre parênteses correspondem ao valor RMS medido.

Configuração			$t_{SP}$ (ms)	$t_{LV}$ (ms)	$t_{total}$ (ms)	Eficiência	
FZ	RD	THL		0,2382(0,0753)	0,0490(0,0163)	0,4058(0,1936)	3804/4942 (76,97%)
		THA	FX	0,2387(0,0756)	0,0530(0,0289)	0,3620(0,1755)	4043/4942 (81,81%)
			DP	0,2404(0,0753)	0,0509(0,0226)	0,3318(0,1825)	4025/4942 (81,44%)
			RS	0,2388(0,0752)	0,0498(0,0234)	0,3458(0,1653)	4022/4942 (81,37%)
		THM	FX	0,2378(0,0753)	0,0571(0,0344)	0,3826(0,1923)	4079/4942 (82,53%)
			DP	0,2390(0,0751)	0,0548(0,0330)	0,3728(0,1841)	4073/4942 (82,41%)
	RS		0,2389(0,0750)	0,0545(0,0327)	0,3603(0,1879)	4073/4942 (82,41%)	
	RV	THL		0,2381(0,0751)	0,0157(0,0111)	0,2972(0,1505)	3781/4942 (76,49%)
		THA	FX	0,2379(0,0752)	0,0123(0,0076)	0,2909(0,1476)	4240/4942 (85,78%)
			DP	0,2382(0,0752)	0,0107(0,0064)	0,2878(0,1121)	4205/4942 (85,08%)
			RS	0,2379(0,0752)	0,0109(0,0070)	0,2870(0,1264)	4203/4942 (85,04%)
		THM	FX	0,2379(0,0753)	0,0278(0,0118)	0,3061(0,1388)	4300/4942 (87,00%)
			DP	0,2379(0,0752)	0,0246(0,0106)	0,2952(0,1405)	4250/4942 (86,00%)
	RS		0,2383(0,0752)	0,0252(0,0126)	0,3026(0,1398)	4247/4942 (85,92%)	
	CB	THL		0,2384(0,0751)	0,0344(0,0220)	0,3463(0,1833)	3791/4942 (76,70%)
		THA	FX	0,2380(0,0753)	0,0255(0,0225)	0,3351(0,1937)	4252/4942 (86,03%)
			DP	0,2384(0,0751)	0,0220(0,0197)	0,3213(0,2076)	4218/4942 (85,33%)
			RS	0,2380(0,0753)	0,0214(0,0208)	0,3177(0,2019)	4218/4942 (85,33%)
		THM	FX	0,2389(0,0752)	0,0266(0,0219)	0,3018(0,1569)	4312/4942 (87,24%)
			DP	0,2393(0,0751)	0,0215(0,0189)	0,2933(0,1549)	4257/4942 (86,12%)
RS	0,2391(0,0752)		0,0181(0,0169)	0,2881(0,1508)	4252/4942 (86,02%)		

Tabela 9.3: Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção por raio de atuação em  $\phi \times z$ . Os valores entre parênteses correspondem ao valor RMS medido.

Configuração			$t_{SP}$ (ms)	$t_{LV}$ (ms)	$t_{total}$ (ms)	Eficiência	
VZ	RD	THL		0,1658 (0,0935)	0,0490(0,0163)	0,3719(0,2426)	3652/4942 (73,89%)
		THA	FX	0,1663 (0,0934)	0,0530(0,0289)	0,3313(0,2340)	3878/4942 (78,46%)
			DP	0,1663 (0,0883)	0,0509(0,0226)	0,3072(0,2558)	3865/4942 (78,20%)
			RS	0,1659 (0,0935)	0,0498(0,0234)	0,3198(0,2417)	3865/4942 (78,20%)
		THM	FX	0, 0,1661 (0,0933)	0,0571(0,0344)	0,3438(0,2411)	3853/4942 (88,08%)
			DP	0,1660 (0,0935)	0,0548(0,0330)	0,3282(0,2426)	4333/4942 (87,67%)
	RS		0,1660 (0,0934)	0,0545(0,0327)	0,3122(0,2665)	4333/4942 (87,67%)	
	RV	THL		0,1661 (0,0935)	0,0157(0,0111)	0,2676(0,2192)	3808/4942 (77,04%)
		THA	FX	0,1656 (0,0935)	0,0123(0,0076)	0,2527(0,2025)	4053/4942 (82,01%)
			DP	0,1661 (0,0932)	0,0107(0,0064)	0,2473(0,2051)	4039/4942 (81,72%)
			RS	0,1661 (0,0933)	0,0109(0,0070)	0,2481(0,2114)	4227/4942 (85,52%)
		THM	FX	0,1658 (0,0936)	0,0278(0,0118)	0,2831(0,2053)	4379/4942 (88,59%)
			DP	0,1661 (0,0934)	0,0246(0,0106)	0,2696(0,2113)	4364/4942 (88,29%)
	RS		0,1661 (0,0934)	0,0252(0,0126)	0,2750(0,2050)	4360/4942 (88,31%)	
	CB	THL		0,1660 (0,0932)	0,0344(0,0220)	0,2954(0,2433)	3652/4942 (73,89%)
		THA	FX	0,1661 (0,0934)	0,0255(0,0225)	0,2792(0,2415)	4260/4942 (86,19%)
			DP	0,1661 (0,0935)	0,0220(0,0197)	0,2582(0,2121)	4252/4942 (86,03%)
			RS	0,1661 (0,0936)	0,0214(0,0208)	0,2573(0,2320)	4252/4942 (86,03%)
		THM	FX	0,1656 (0,0934)	0,0266(0,0219)	0,3221(0,2371)	4403/4942 (89,09%)
			DP	0,1658 (0,0934)	0,0215(0,0189)	0,3104(0,2225)	4348/4942 (87,98%)
RS	0,1657 (0,0934)		0,0181(0,0169)	0,3037(0,2355)	4343/4942 (87,87%)		

Tabela 9.4: Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários pelo método de seleção por vizinhança. Os valores entre parênteses correspondem ao valor RMS medido.

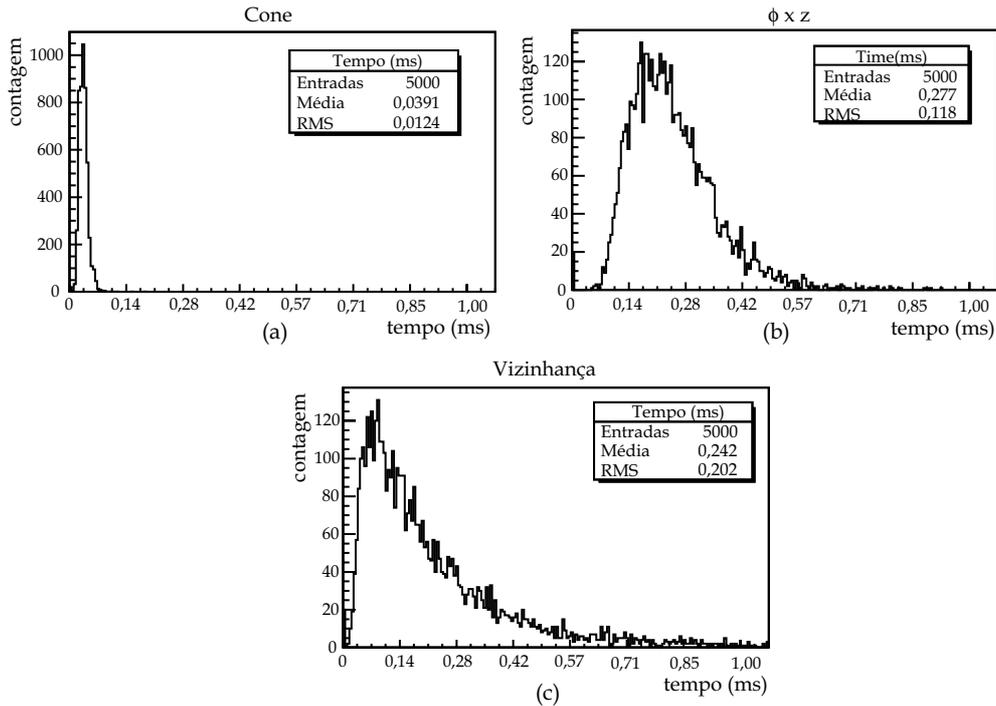


Figura 9.3: Tempo de execução do sistema de seleção geométrica de pontos, em milissegundo. Eventos de alta luminosidade.

A Figura 9.4 apresenta os histogramas de tempo de processamento dos métodos de seleção geométrica de pontos para eventos de baixa luminosidade. A comparação dos histogramas de eventos de alta luminosidade (Figura 9.3) com os de baixa luminosidade (Figura 9.4) mostra que a seleção de pontos é três vezes mais rápida em eventos de baixa luminosidade. Isto se deve ao fato dos eventos de alta luminosidade possuírem mais pontos que os eventos de baixa luminosidade.

Os gráficos das Figuras 9.5 e 9.6 ilustram o comportamento dos sistemas de seleção de pontos com relação ao tempo de processamento. O eixo das abscissas representa o tempo de processamento e o eixo das ordenadas apresenta o número de pontos em cada evento. O método de seleção por cone possui um comportamento linear de tempo de processamento com relação ao número de pontos. Os métodos de raio  $\phi \times z$  e vizinhança possuem um comportamento exponencial com relação ao número de pontos nos eventos. O gráfico representado pela letra (d) em ambas as Figuras ilustra o tempo de processamento do método *ZFinder* com relação ao número de pontos.

De forma similar aos histogramas de tempo de processamento dos métodos

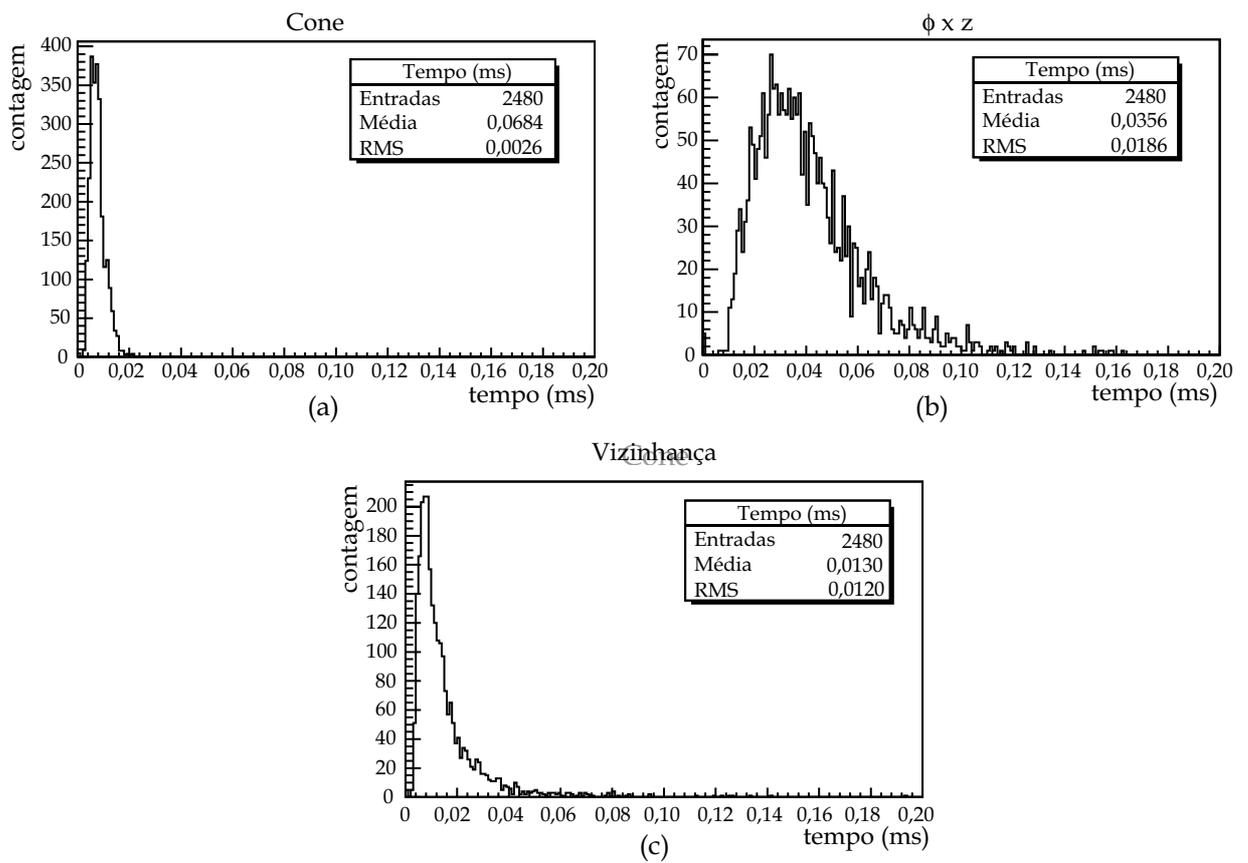


Figura 9.4: Tempo de execução do sistema de seleção geométrica de pontos, em milisegundo. Eventos de baixa luminosidade.

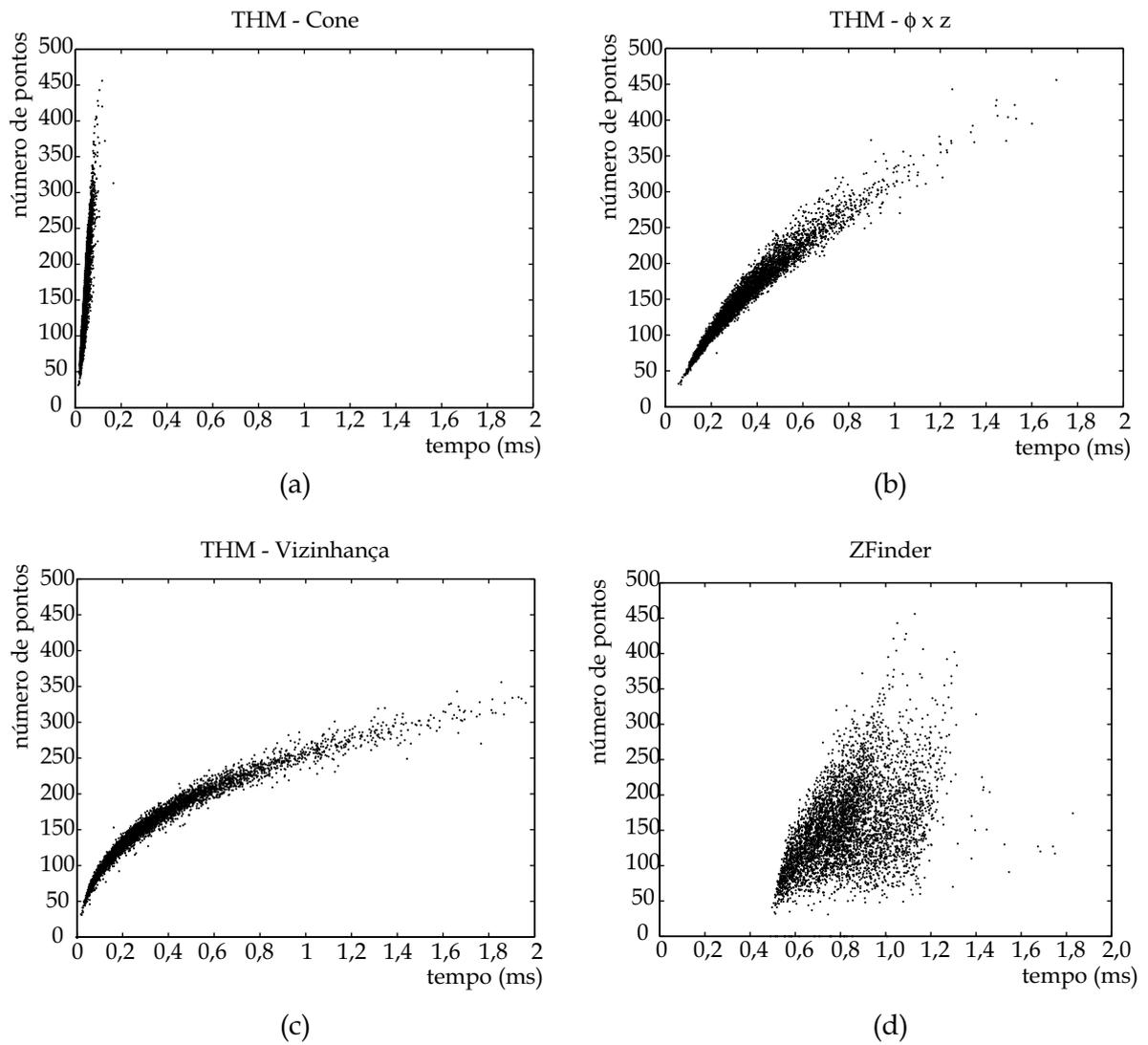


Figura 9.5: Tempo de processamento com relação ao número de pontos por evento em alta luminosidade.

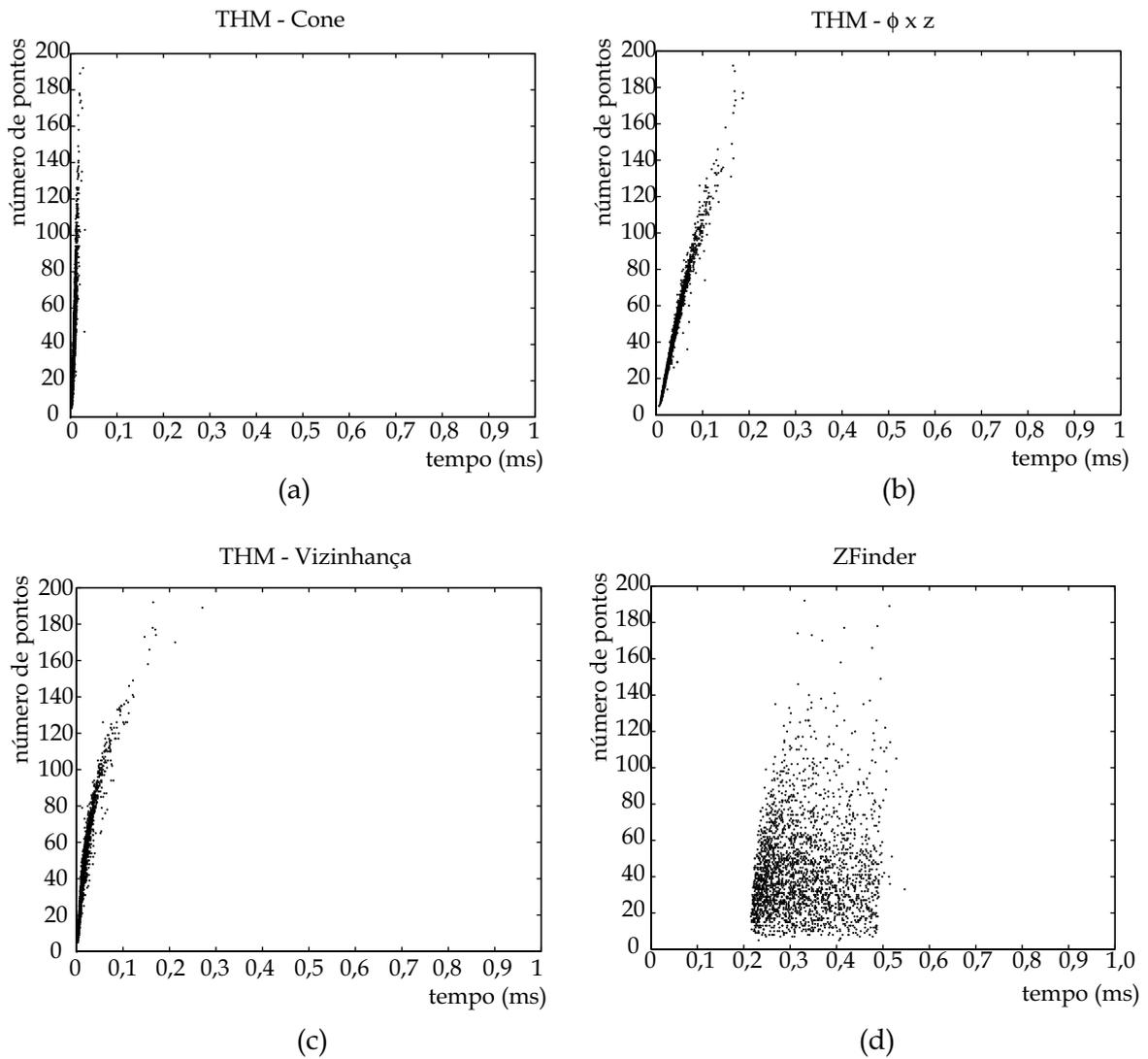


Figura 9.6: Tempo de processamento com relação ao número de pontos por evento em baixa luminosidade.

de seleção de pontos, foram realizados histogramas dos tempos de processamento do sistema de localização de vértices. Os histogramas das Figuras 9.7 e 9.8 mostram que não há variação significativa nos tempos das transformadas para amostras de baixa e alta luminosidade, uma vez que a média para eventos de baixa luminosidade é de  $0,009 (\pm 0,003)$  ms e para eventos de alta luminosidade é de  $0,049 (\pm 0,015)$  ms.

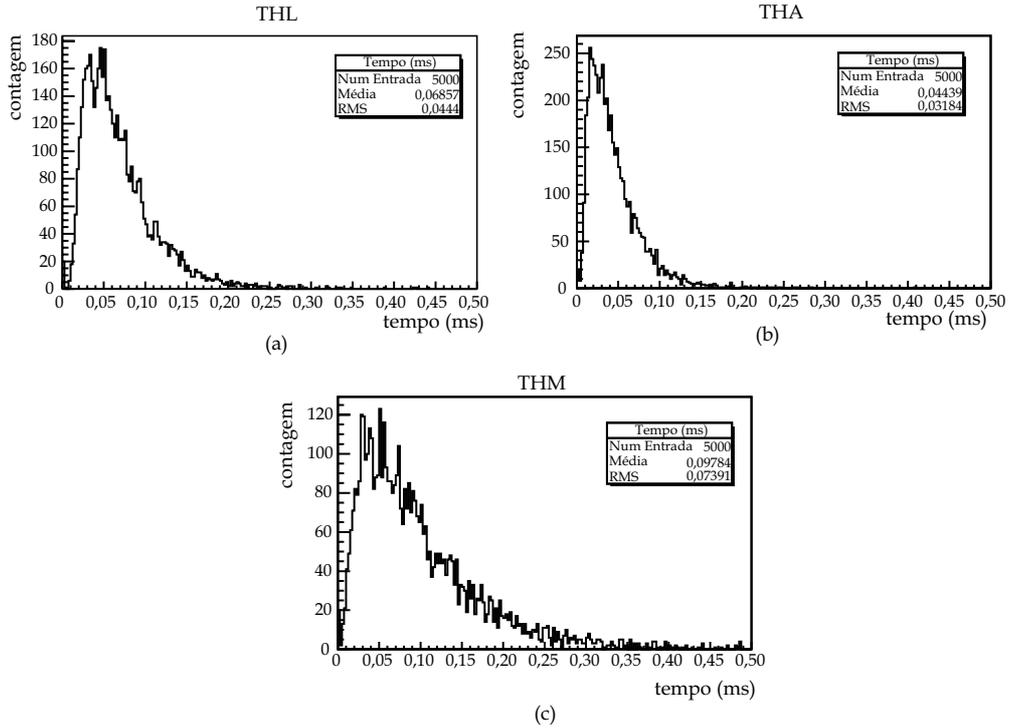


Figura 9.7: Tempo de execução do sistema de localização para a configuração de escolha de pontos por  $\rho$  distintos e número de recursão fixa. Eventos de alta luminosidade.

Os histogramas da Figura 9.9 representam a diferença entre os valores de simulação dos vértices ( $z_{true}$ ) e os obtidos pelo algoritmo de seleção de pontos por vizinhança, escolha de pontos pelo método centro bordas, transformada de Hough morfológica com condição de parada fixa (VZCBTHMFX,  $z_{vtx}$ ). O valor médio do histograma para amostras de baixa luminosidade consiste em  $0,03204$  mm com um RMS de  $0,2560$  mm. Para alta luminosidade o valor médio é de  $0,04793$  mm com um RMS de  $0,3791$  mm. A eficiência de acerto para eventos de baixa luminosidade consiste em  $2382/2480$  (96,04%), e para eventos de alta luminosidade consiste em  $4403/4942$  (89,09%).

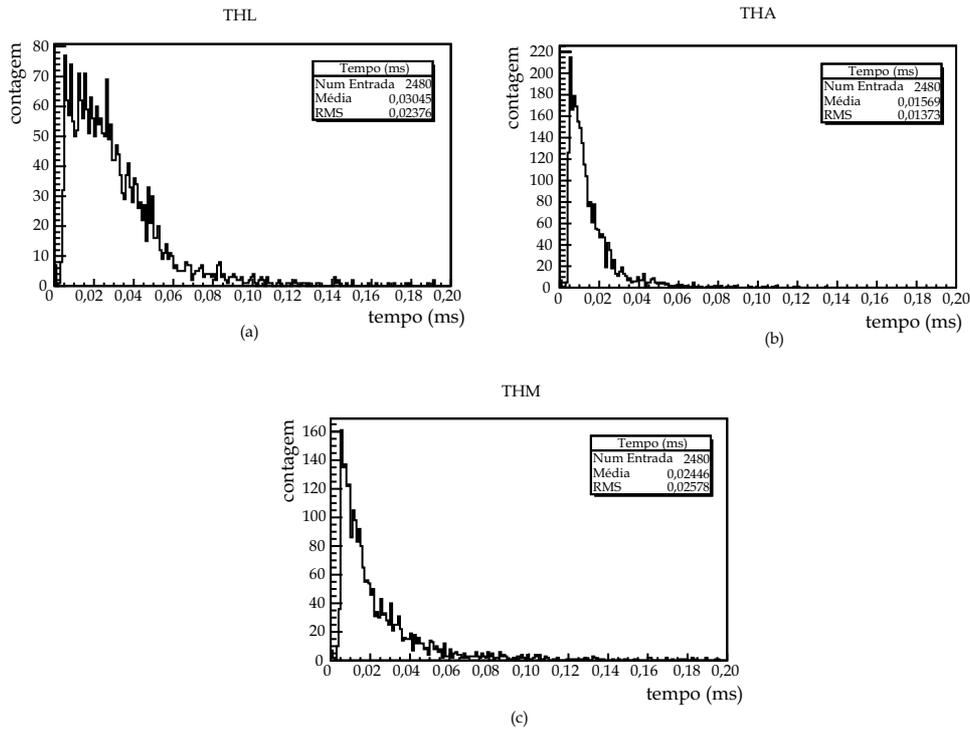
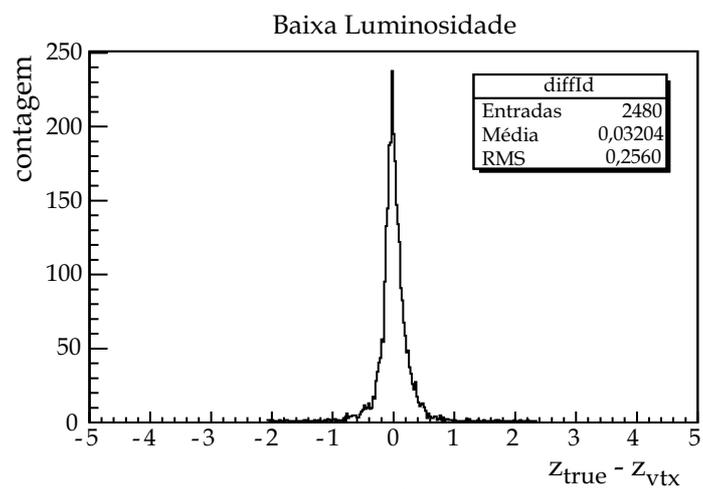


Figura 9.8: Tempo de execução do sistema de localização para a configuração de escolha de pontos por  $\rho$  distintos e número de recursão fixa. Eventos de baixa luminosidade.

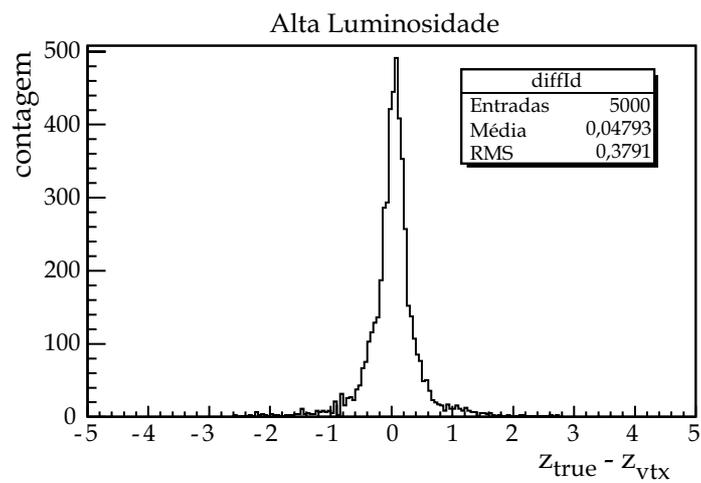
Os gráficos e histogramas desta seção apresentaram o comportamento do sistema de localização de vértices configurado da seguinte forma: escolha de pontos pelo método centro bordas e transformada de Hough morfológica com número de recursões fixa. Esta configuração aliada com o sistema de seleção por vizinhança obteve a maior eficiência, de acordo com a Tabela 9.4. Foram apresentados os gráficos dessa configuração de localização de vértices com todas as variações do sistema de seleção de pontos (cone, raio  $\phi \times z$ , vizinhança), onde foi observado um comportamento exponencial do tempo de execução com relação ao número de pontos para os sistemas de raio  $\phi \times z$  e de vizinhança.

## 9.2 Reconstrução de Vértices Primários no Ambiente de Desenvolvimento ATHENA

O teste realizado no ambiente de desenvolvimento ATHENA utilizou arquivos de elétrons individuais de 25 GeV à baixa luminosidade e com empilhamento. Uma



(a)



(b)

Figura 9.9: Diferença entre os valores de simulação ( $z_{true}$ ) e os valores obtidos pelo VZCBTHMFX ( $z_{vtx}$ ) para amostras de baixa (a) e alta (b) luminosidade.

vez que o ambiente ATHENA executa toda a cadeia do sistema de filtragem do ATLAS, os eventos são gerados pela simulação Monte Carlo, seguida da simulação da resposta dos detectores e do sistema de filtragem LVL1 e, por fim, a entrada do sistema de localização de vértice primário consiste na imagem volumétrica selecionada pelo sistema de Selecionamento de Regiões. Os eventos são, portanto, gerados durante o processo de simulação do sistema de filtragem do ATLAS. Desta forma, os eventos utilizados no ambiente ATHENA são diferentes daqueles utilizados para teste e validação em separado do sistema de reconhecimento de vértices por transformada de Hough.

A tabela 9.5 apresenta os tempos de execução e a eficiência de algumas configurações de sistemas de reconstrução de vértices. Como as amostras utilizadas no ATHENA são de baixa luminosidade, as colunas  $t_{total}$  e Efic apresentam o tempo de execução e a eficiência para as amostras de validação de baixa luminosidade do conjunto de dados.

O algoritmo de referência *ZFinder* obteve um tempo de execução de  $0,3823(\pm 0,1847)$  ms e uma eficiência de  $1730/1800$  (96,11%). A Tabela 9.5 mostra que os tempos de execução no ATHENA são equivalentes aos tempos medidos no sistema *stand alone*. A eficiência do ATHENA mostrou ser superior para todos os sistemas de configuração. Isto se deve ao fato dos eventos possuírem menos efeito de empilhamento e por conseqüência menos nível de ruído.

A maior eficiência foi verificada para a configuração VZCBTHMFX, assim como observado no sistema *stand-alone*. Os tempos de execução das configurações propostas foram no mínimo 8 vezes inferior ao do algoritmo referência *ZFinder*.

### 9.3 Reconstrução de Vértices Primários no SHARC ADSP-21160

Conforme apresentado na Seção 8.6, foi realizado um teste do sistema de reconstrução de vértices primários por transformada de Hough no kit de desenvolvimento SHARC ADSP-21160 EZ-KIT Lite[86]. O algoritmo testado foi o de seleção de pontos por vizinhança, escolha dos pares pelo critério centro bordas, transformada de Hough adaptativa com critério de parada por diferença entre pi-

Configuração				$t_{total}$ (ms)	Efic	$t_{ATHENA}$	Efic ATHENA
CN	RD	THA	FX	0,0158(0,0061)	1374/2480(55,40%)	0,0143(0,0054)	1345/1800 (74,72%)
CN	RV	THA	FX	0,0162(0,0061)	1224/2480 (49,35%)	0,0172(0,0052)	1312/1800 (72,88%)
FZ	RD	THL		0,0467(0,0224)	1909/2480 (76,97%)	0,0437(0,0231)	1523/1800 (84,61%)
FZ	RD	THA	FX	0,0446(0,219)	2029/2480 (81,81%)	0,0429(0,185)	1654/1800 (91,88%)
FZ	RV	THA	DP	0,0453(0,203)	2110/2480 (85,08%)	0,0438(0,193)	1665/1800 (92,50%)
VZ	RD	THL		0,0241(0,0158)	1879/2480 (75,76%)	0,0183(0,0160)	1515/1800 (84,16%)
VZ	RV	THA	FX	0,0221(0,0153)	1946/2480 (78,46%)	0,0193(0,0168)	1579/1800 (87,72%)
VZ	CB	THA	DP	0,0218(0,0143)	2134/2480 (86,04%)	0,0189(0,0179)	1704/1800 (94,66%)
VZ	CB	THM	FX	0,0250(0,0157)	2382/2480(96,04%)	0,0203(0,0165)	1776/1800 (98,66%)

Tabela 9.5: Tempos de processamento e eficiências das configurações dos sistemas de reconstrução de vértices primários testadas no ambiente ATHENA. Os valores entre parênteses correspondem ao valor RMS medido.

cos (VZCBTHADP), pois na ocasião do teste, esse método consistia no sistema de melhor eficiência.

A Figura 9.10 apresenta os tempos de processamento das recursões da THA e do tempo total de processamento. De modo a fazer as comparações de tempo compatíveis com os tempos medidos no PC, os resultados apresentados possuem seus valores escalados para o ADSP-BF533SKBC750, que possui 756 MHz de velocidade de processamento, sendo o DSP mais recente da Analog Devices. Isto se deve ao fato dos tempos medidos no PC serem escalados para uma máquina de 4GHz, correspondendo à previsão de processador a ser lançado no ano de funcionamento do experimento ATLAS em 2007.

A recursão 0 possui o maior tempo de processamento, conforme esperado. Foram utilizadas amostras de alta luminosidade, as mesmas utilizadas na versão *stand-alone*. Foi verificado um tempo total de processamento de 0,32 ms com valor RMS de 0,35 ms, sendo compatível com os tempos de execução desse algoritmo no ATHENA (0,2501(0,202) ms) e na versão *stand-alone* (0,2582(0,2121) ms).

A Figura 9.11 apresenta o gráfico de variação do tempo de execução com relação ao número de pares ( $m, c$ ) dos eventos. Há uma variação linear entre esses dois parâmetros, diferente do observado na execução no PC onde o comportamento foi exponencial. Isso ocorreu devido ao fato do teste realizado no DSP utilizar

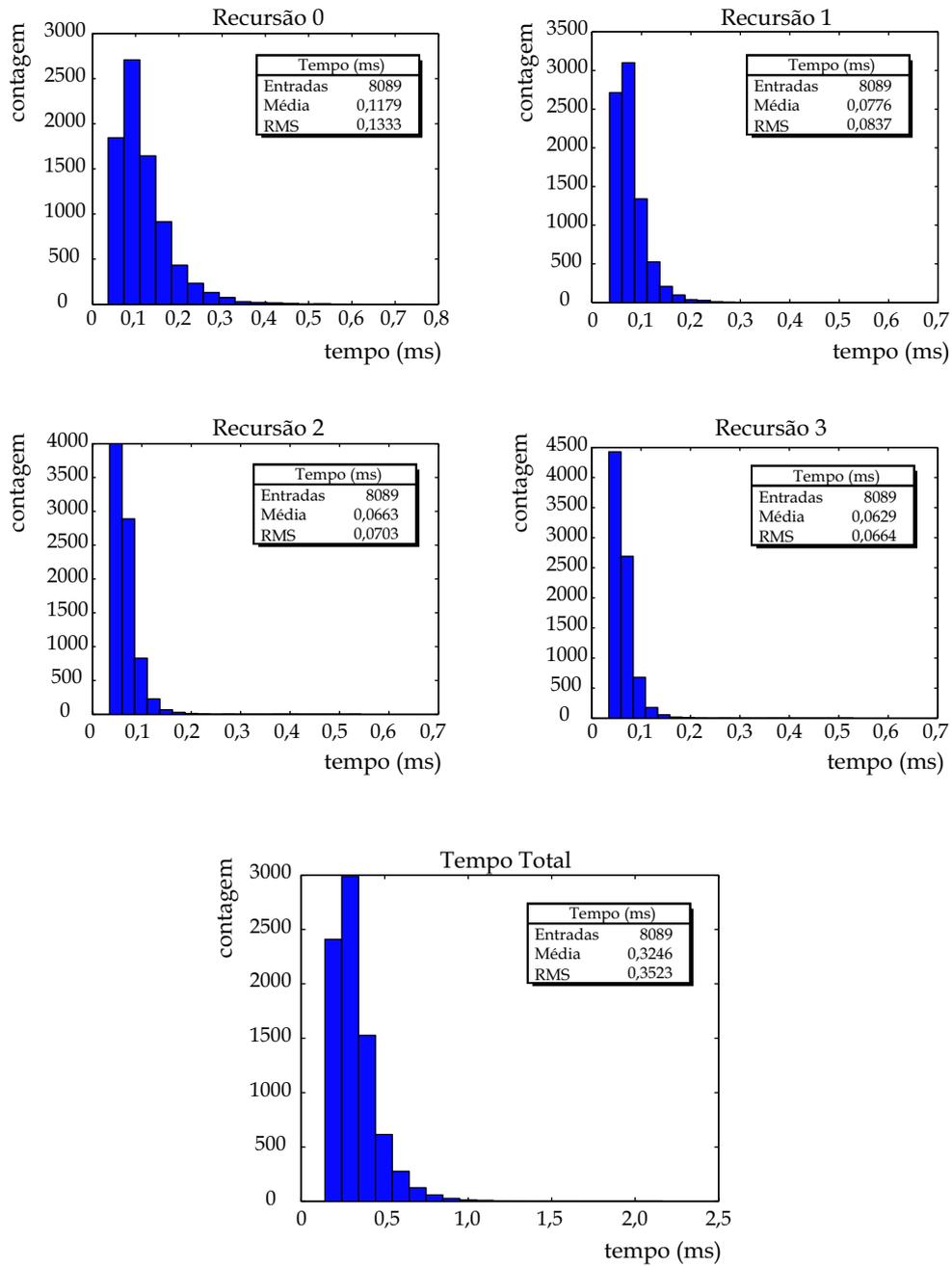


Figura 9.10: Tempo em ms de execução no ADSP-BF533SKBC750. (a) recursão 0; (b) recursão 1; (c) recursão 2; (d) recursão 3; (e) tempo total.

somente os pares  $(m, c)$  da transformada, ou seja, só ter sido testado no DSP o sistema de recursão e de condição de parada do algoritmo VZCBTHADP. Desta forma, o comportamento exponencial observado no PC ocorre devido ao módulo de seleção de pontos e não de localização do vértice primário. No gráfico da Figura 9.11 observa-se eventos com mais de 5000 pontos. Isto ocorre pois o eixo das ordenadas corresponde aos pares  $(m, c)$  do espaço de parâmetros de Hough.

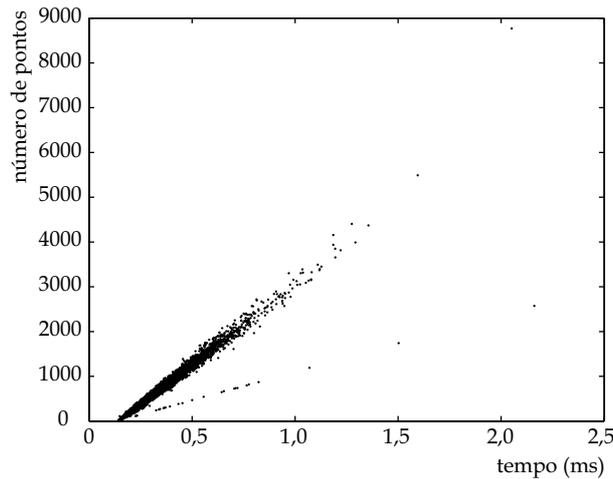


Figura 9.11: Gráfico do tempo de execução com relação ao número de pontos nos eventos.

## 9.4 Considerações

Foi observado que a eficiência do sistema de reconstrução de vértices primários por transformada de Hough encontra-se compatível com a do algoritmo referência *ZFinder* tanto para amostras de alta quando baixa luminosidade. Os tempos de processamento, no entanto, são de 3 a 15 vezes inferiores ao algoritmo referência.

A vantagem do ganho em tempo de processamento sem perdas na precisão de localização é bem positiva em um ambiente de execução em tempo real, como no caso desse trabalho. A latência desse sistema de reconstrução de vértices para o LVL2 é de 1 ms. O sistema de reconstrução por transformada de Hough possui um tempo de processamento de 0,0203(0,0165) ms para eventos de baixa luminosidade e de 0,3221(0,2371) ms para eventos de alta luminosidade<sup>1</sup>. O sistema de localização

---

<sup>1</sup>Ambos os tempos correspondem ao algoritmo de maior eficiência, o VZCBTHMFX

*ZFinder* possui tempos de 0,322(0,007) ms para amostras de baixa luminosidade e 0,818(0,169) ms para eventos de alta luminosidade.

O sistema de reconstrução por transformada de Hough é tão preciso quanto o *ZFinder*, uma que a eficiência para amostras de baixa luminosidade é de  $\approx 96,04\%$  e eventos de alta luminosidade é  $\approx 89,09\%$ . O *ZFinder* possui eficiências de  $\approx 94,43\%$  para eventos de baixa luminosidade e  $\approx 88,14\%$  para eventos de alta luminosidade. É importante ressaltar que as eficiências foram calculadas para o mesmo conjunto de eventos, tanto em baixa quanto em alta luminosidade.

Alguns sistemas de reconstrução por transformada de Hough foram testados no ambiente de desenvolvimento ATHENA, com o objetivo de verificar o real funcionamento dos algoritmos. Foi utilizado um conjunto de dados diferente daqueles de teste e validação dos sistemas. Foram observados tempos de execução e eficiências equivalentes às medidas no sistema *stand-alone*.

Uma versão do sistema de transformada de Hough também foi testada no ADSP-21160 da Analog Devices. Foram observados tempos de processamento e eficiências equivalentes às aquelas mensuradas no PC.

# Capítulo 10

## Conclusão

Esse trabalho apresentou a utilização da transformada de Hough no processo de reconstrução online de vértices primários de colisão no ambiente de alta taxa de eventos do LHC. Os métodos de transformada de Hough foram empregados na construção de um sistema *online* de processamento de imagens volumétricas de colisões de partículas nos detectores de traço do ATLAS, um dos laboratórios de detecção posicionado em um dos pontos de colisão do LHC.

O desenvolvimento dos métodos de seleção de regiões e de transformada de Hough foram realizados no ambiente de emulação de sistemas de filtragem do ATLAS, o ATHENA. Ambos os métodos foram projetados para operar no ambiente de filtragem de segundo nível do ATLAS, ou HLT. Este nível de filtragem possui latência de execução de 10 ms para a reconstrução completa do evento. Desta forma, como a cadeia de reconstrução contém mais de um algoritmo, é requerido um tempo de execução menor que 1 ms para cada algoritmo da cadeia de reconstrução.

A primeira etapa do trabalho consistiu no desenvolvimento da ferramenta de seleção de regiões (*Region Selector*) para o sistema de filtragem do ATLAS. Atualmente, o sistema de seleção de regiões dá suporte a todos os subdetectores do ATLAS e é utilizado pelos sistemas HLT e de Reconstrução do ATLAS. Recentemente, o algoritmo e suas documentações foram julgados por uma comissão de avaliação do ATLAS, formada por quatro especialistas do CERN. O documento de avaliação apontou a importância desta ferramenta de seleção para os sistemas de filtragem do ATLAS e propôs algumas ampliações do algoritmo para complementação do trabalho realizado.

Atualmente o seletor de regiões apresenta uma seleção que leva em consideração a incerteza substancial na posição  $z$  do vértice primário devido ao espalhamento do feixe LHC. A seleção por cone desenvolvida ao longo das diferentes versões do sistema apresenta uma extensão em  $\eta$  e  $\phi$  levando, desta forma, em consideração o espalhamento do vértice  $z$ . A dependência radial  $\Delta\eta$  apresentada pela seleção por cones reduz o ruído de fundo, principalmente nos detectores de traço mais externos onde o cone é mais estreito.

A etapa de localização do vértice de colisão iniciou-se pela utilização da transformada de Hough em suas formas local e adaptativa para a reconstrução do vértice primário de elétrons individuais sem empilhamento. Ao longo do desenvolvimento, a transformada de Hough morfológica mostrou uma melhor eficiência e um tempo de processamento adequado ao que se encontra disponível no segundo nível de *trigger*. Esta transformada morfológica possui as características da transformada adaptativa, devido às recursões no espaço de parâmetros, aliadas com a exploração da morfologia do espaço de parâmetros. Foi observado que a transformada de Hough Morfológica possui eficiência e tempo de execução melhores que os do algoritmo *ZFinder*, base de projeto atual do ATLAS.

Os tempos de processamento desta transformada são de  $0,3221(\pm 0,2371)$  ms para eventos de alta luminosidade e de  $0,0203(\pm 0,0165)$  ms para eventos de baixa luminosidade. As eficiências são de  $\approx 89,09\%$  e de  $\approx 96,04\%$  para eventos de alta e baixa luminosidade, respectivamente. O algoritmo *ZFinder* para o mesmo conjunto de dados apresentou eficiências de  $\approx 88,14\%$  e  $\approx 94,43\%$ . O tempo de execução do *ZFinder* foi de  $0,322(\pm 0,007)$  ms e de  $0,818(\pm 0,169)$  ms.

A transformada de Hough mostrou ser um método alternativo para a localização do vértice primário de colisão para eventos de elétrons com empilhamento. Este método proporciona um ganho de 3 a 15 vezes no tempo de processamento, com uma precisão ligeiramente superior à do *ZFinder*.

## 10.1 Trabalhos Futuros

Algumas possíveis extensões desse trabalho já se encontram identificadas. A primeira consiste em investigar o método de seleção por cone, uma vez que este

método possui um tempo de execução bastante rápido. Um estudo de variação no raio da base do cone pode aumentar de forma significativa a eficiência do sistema.

Outra possível extensão consiste no estudo das condições de parada adaptativa por resolução e diferença entre picos. Um ajuste fino dessas condições de parada pode ser realizado objetivando aumentar a precisão de saída do sistema.

# Referências Bibliográficas

- [1] A. BADALÀ, R. B. E. A., *Full 3D Event-by-Event Vertex Reconstruction in ALICE by Pixel Layers*. ALICE Internal Note-ITS, May 2001. ALICE/INT-2001-26.
- [2] COMMUNITY, A. I. D., *Inner Detector Technical Design Report*, v. I. ATLAS CERN, Abril 1997. ISBN = 92-9083-102-2.
- [3] P. BILLOIR, S. Q., “Fast vertex fitting with a local parametrization of tracks”. In: *Nucl. Instrum. Methods Phys. Res., A 311, 139-150*, 1992.
- [4] R. FRÜHWIRTH, T. S., “A Gaussian-Sum Filter for Vertex Reconstruction”. In: *9th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2003)*, Tsukuba, Japan, 1-5, pages 217-221, December 2003.
- [5] E. CHABANAT, N. E., “Deterministic Annealing for Vertex Finding at CMS”. In: *Computing in High Energy Physics and Nuclear Physics (CHEP 2004)*, Interlaken, Switzerland, pages 287, October 2004.
- [6] W. HOOGLAND, J. VISSCHERS, L. W. N. Z., “Status report on the development of the primary vertex fast reconstruction”. In: *NIKHEF*, 1997.
- [7] GEORGE, S., “Design of the PESA core software Data Manager”. In: *EDMS Id:390533, ATL-DH-EN-0004*, 2005. Internal Report.
- [8] L. M. C. S. RODRIGUES, A. C. DOS REIS, E. M. F. C., TERRA, J., JR., N. A., *Anais da IV Escola do CBPF*. CBPF, 2002. capítulo: Introdução à Física de Partículas, página: 219-245.

- [9] BOIXADER, G., SOUTHWORTH, B., *The world of particles*, n. 67 páginas. CERN Desktop Publishing, 1991. ISBN = 0-07-004473-2.
- [10] H. F. SCHOPPER, S. A. K., KRIGE, J., “CERN’s Early History Revisited”. In: *Phys. Today*, p. 87, April 2005.
- [11] BLUM, W., ROLANDI, L., *Particle Detection with Drift Chambers*. Springer, 1993.
- [12] BEISER, A., *Concepts of Modern Physics*, n. 88-90. McGraw-Hill, 1987. ISBN = 0-07-004473-2.
- [13] LEWIS, J., GROOM, C., “A Aventura das Partículas”, <http://www.aventuradasparticulas.ift.unesp.br/>, Fevereiro 2005.
- [14] DAWSON, S., “Introduction to the physics of Higgs bosons”. In: *Theoretical Advanced Study Institute in Elementary-particle Physics*, pp. 445–505, November 1994.
- [15] ANJOS, A. R. D., “Sistema Neronal Rápido de Decisão Baseado em Calorimetria de Altas Energias”, Abril 2001.
- [16] DOMENICI, D., *Detection of muons in the LHCb experiment: the aging of RPC detectors and the study of  $Z^0 \rightarrow \mu^+\mu^-$* . Ph.D. dissertation, D.Sc Thesis, Università degli Studi di Roma, 2003.
- [17] “Virtual reality image of ATLAS with cavern walls visible”, ATLAS Photo Collection, ATL-PHO-GEN-2002-002, 2002, Computer Graphics Photograph.
- [18] “The ATLAS Experiment Tour”, [http://atlas.ch/etours\\_exper/etours\\_exper07.html](http://atlas.ch/etours_exper/etours_exper07.html), Agosto 2005.
- [19] COLLABORATION, A., *Muon Spectrometer Technical Progress Report*. Technical Report, ATLAS CERN, June 1997.
- [20] MOSER, H. G., “The status of the ATLAS inner detector”. In: *9th International Conference on B Physics at Hadron Machines Beauty 2003*, Oct 2003. Pittsburgh, PA, USA, pages 99-103.

- [21] BENEKOS, N. C. E. A., “B-physics performance with Initial and Complete Inner detector layouts in Data Challenge-1”. In: *Internal Report, ATLCOM-PHYS-2004-073, CERN-ATL-COM-PHYS-2004-073, ATL-PHYS-2005-002*, 2004.
- [22] HÜGGING, F. G., “The ATLAS Pixel Detector”. In: *2004 IEEE Nuclear Science Symposium And Medical Imaging Conference NSS-MIC*, Oct 2004.
- [23] SANDAKER, H., “The readout and control system for the ATLAS Semiconductor Tracker”. In: *ATLAS Notes, Internal Report*, May 2004.
- [24] J. D. PLUMMER, M. D. DEAL, P. B. G., *Silicon VLSI Technology: Fundamentals, Practice, and Modeling*. Prentice Hall, 1 edition, 2000. ISBN = 0130850373.
- [25] COLLABORATION, A., *First-Level Trigger Technical Design Report*. Technical Report, ATLAS CERN, June 1998. CERN/LHCC/98-14.
- [26] COLLABORATION, A., *DAQ, EF, LVL2 and DCS Technical Progress Report*. Technical Report, ATLAS CERN, June 1998. CERN/LHCC/98-16.
- [27] INTERNAL REPORT, A. C., *Trigger Performance Status Report*. ATLAS CERN, June 1998. CERN/LHCC/98-15.
- [28] ATLAS, *ATLAS Detector Physics Performance Technical Design Report*, v. I. ATLAS CERN, May 1999.
- [29] TECHNICAL REPORT, A. C., *Trigger Performance Technical Progress Report*. ATLAS CERN, 1999.
- [30] COLLABORATION, A., *ATLAS Detector and Physics Performance Technical Design Report*. ATLAS CERN, 1999. CERN/LHCC/99-14.
- [31] GROUP, A. C., *ATHENA User Guide and Tutorial*. Users Guide, ATLAS CERN, August 2001.
- [32] DAMERELL, C., “Exceptional Colloquium: Vertex Detectors and the Linear Collider”. In: *CERN Colloquium*, CERN, April 2005.

- [33] J. C. HART, D. H. S., “Track and vertex fitting in an inhomogeneous magnetic field”. In: *Rutherford Appleton Lab. Chilton*, 1983.
- [34] G. N. PATRICK, B. S., “Vertex fitting of several helices in space”. In: *Internal Report, CERN, Geneva*, 1985.
- [35] FRÜHWIRTH, R., “Application of Kalman filtering to track and vertex fitting”. In: *Nucl. Instrum. Methods Phys. Res., A 262, 444*, 1987.
- [36] R. LUCHSINGER, C. G., “Vertex reconstruction by means of the method of Kalman filter”. In: *Nucl. Instrum. Methods Phys. Res., A*, 1992.
- [37] P. BILLOIR, R. FRÜHWIRTH, M. R., “Track element merging strategy and vertex fitting in complex modular detectors . - rev. version”. In: *Österr. Akad. Wiss. Inst. Hochenergiephys*, 1985.
- [38] YATSUNENKO, Y. A., “Vertex reconstruction without track reconstruction”. In: *Joint Inst. Nucl. Res.*, 1989.
- [39] KARIMÄKI, V., “Effective vertex fitting for particle physics experiments”. In: *Helsinki Inst. Phys.*, 1997.
- [40] KOSTYUKHIN, V., “VKalVrt - package for vertex reconstruction in ATLAS”. In: *ATLAS Internal Report, Geneva, CERN*, 2003.
- [41] W. A. MITAROFF, G. RICHTER, W. W., “Detector-independent vertex reconstruction toolkit (VERTIGO)”. In: *Computing in High Energy Physics and Nuclear Physics (CHEP 2004), Interlaken, Switzerland , pages 272*, October 2004.
- [42] SAXON, D. H., “Three-dimensional track and vertex fitting in chambers with stereo wires”. In: *RAL*, 1984.
- [43] P. BILLOIR, S. Q., “Further test for the simultaneous pattern recognition and track fitting by the Kalman filtering method”. In: *Nucl. Instrum. Methods Phys. Res., A 295, 492-500*, 1990.
- [44] FRÜHWIRTH, R., “Track and vertex fitting”. In: *12th CERN School of Computing , Bad Herrenalb, Germany, pages 217-235*, 1990.

- [45] R. FRUHWIRTH, P. K. E. A., “Vertex reconstruction and track bundling at the LEP collider using robust algorithms”. In: *Comput. Phys. Commun.* 96, 189-208, 1996.
- [46] KARIMÄKI, V., “Effective Vertex Fitting”. In: *CERN, Geneva*, p. 51, 1997. CMS Note, HIP-1997-77 EXP.
- [47] DANTAS, A., “Transformada de Hough para Detecção de Traços em um Sistema Rastreador de Física de Partículas”, Abril 2001.
- [48] A. WILDAUER, F. A., “Vertex finding and B-tagging algorithms for the ATLAS Inner Detector”. In: *Computing in High Energy Physics and Nuclear Physics (CHEP 2004), Interlaken, Switzerland*, October 2004. pages 419.
- [49] N. KONSTANTINIDIS, H. D., “Determination of the  $z$  position of primary interactions in ATLAS prior to track reconstruction”. In: *ATL-DAQ-2002-014*, July 2002.
- [50] P. BILLOIR, S. Q., “Simultaneous pattern recognition and track fitting by the Kalman filtering method”. In: *Nucl. Instrum. Methods Phys. Res., A* 294, 219-228, 1990.
- [51] FRÜHWIRTH, R., “Track fitting with non-Gaussian noise”. In: *Comp. Phys. Comm.*, 100, 1, 1997.
- [52] KITAGAWA, G., “Non-Gaussian seasonal adjustment”. In: *Comput. Math. Appl.*, 18, 503, 1989.
- [53] “ORCA”, <http://cmsdoc.cern.ch/orca/>, Agosto 2005.
- [54] R. FRÜHWIRTH, W. W. E. A., “New vertex reconstruction algorithms for CMS”. In: *Geneva, CERN*, June 2003.
- [55] N. KONSTANTINIDIS, S. STEFANIDIS, M. R. S., “Optimisation of the size of the EmTau Region of Interest for the ATLAS Level-2 Trigger using the Eletromagnetic Calorimeter”. In: *Nota técnica do ATLAS no. ATLAS-DAQ-PUB-2005-002, 15 páginas*, October, 2005.

- [56] KALMAN, R. E., “A new approach to linear filtering and prediction problems”. In: *J. Basic Eng.* 82, 34, 1961.
- [57] FRUHWIRTH, R., *Application of Filter Methods to the Reconstruction of Tracks and Vertices on Events of Experimental High Energy Physics*. Ph.D. dissertation, PhD Thesis, HEPHY-PUB 516/88, Institut für Hochenergiephysik der Österreichischen Akademie der Wissenschaften, Vienna, 2003.
- [58] BILLOIR, P., In: *Comput. Phys. Commun.* 57, 390, 1989.
- [59] DENBY, B., “Neural networks and cellular automata in experimental high energy physics”. In: *Comput. Phys. Commun.* 49, 429, 1988.
- [60] PETERSON, C., “Track finding with neural networks”. In: *Nucl. Instrum. and Meth. A* 279, 537, 1989.
- [61] DUDA, R. O., HART, P. E., “Pattern Classification and Scene Analysis”. In: *Wiley, New York*, 1973.
- [62] GROTE, H., “Pattern recognition in high energy physics”. In: *Rep. Prog. Phys.* 50, 473, 1987.
- [63] OHLSSON, M., PETERSON, C., “Track finding with deformable templates”. In: *Computer Physics Communication*, 71, 77-98, 1992.
- [64] M. OHLSSON, C. P., “Track finding with deformable templates - the elastic arms approach”. In: *Computer Physics Communication*, pp. 77-98, 1992.
- [65] A. YUILLE, K. H., PETERSON, C., “Particle tracking by deformable templates”. In: *IEEE International Joint Conference on Neural Networks. Seattle, WA. vol. 1 p.7*, 1991.
- [66] KIN, P., SIU, W. C., “Object recognition with a 2-D Hough domain”. In: *IEEE International Symposium on Circuits and Systems, ISCAS, Digital object identifier: 10.1109/ISCAS.1994.409087, pages 5-8, vol. 3*, 1994.
- [67] CHANG, S., HORNG, J. H., “Determination of Circular Arc Length and Midpoint by Hough Transform”. In: *IEEE International Symposium on Circuits and Systems, ISCAS*, 1994.

- [68] C. SZE-TSAN, P. K., SIU, W. C., “Peak detection in Hough transform via self-organizing learning”. In: *IEEE International Symposium on Circuits and Systems, ISCAS, 139-143*, 1995.
- [69] HOUGH, P. V. C., “Methods and means for recognizing complex patterns”. In: *U.S. Patent 3 069 654*, Dezembro 1992.
- [70] J. ILLINGWORTH, J. K., “A survey of the Hough transform”. In: *Computer Vision, Graphics and Image Processing, v. 44, pp. 87-116*, 1988.
- [71] JOSE MANOEL DE SEIXAS, A. C. H. D., “Hough Transform for a Track Recognition System in Particle Physics Experiments”. In: *Second IEEE South-American Workshop on Circuits and Systems, Rio de Janeiro. Proc. Second IEEE South-American Workshop on Circuits and Systems.*, 2001.
- [72] ALINE DA ROCHA GESUALDI, JOSE MANOEL DE SEIXAS, G. M. R. H. M. P. A., “Localização de Vértices Primários de Colisão do ATLAS por Transformada de Hough”. In: *XXVI - Encontro Nacional de Física de Partículas e Campos, São Lourenço.*, 2005.
- [73] J. ILLINGWORTH, J. K., “The adaptive Hough transform”. In: *Trans. Pattern Analysis and Machine Intelligence, v.9, n.5, pp. 690-697*, 1987.
- [74] SCHMIDT, B., SCHIMMLER, M., SCHRÖDER, H., “Morphological Hough Transform on the Instruction Systolic Array.” In: *Euro-Par*, pp. 798–806, 1997.
- [75] SIM, L. C., SCHRÖDER, H., LEEDHAM, G., “Major line removal morphological hough transform on a hybrid system.”, *J. Parallel Distrib. Comput.*, v. 64, n. 9, pp. 1060–1068, 2004.
- [76] A C H DANTAS, JOSE MANOEL DE SEIXAS, P. T. C., “Novos Algoritmos Baseados na transformada de Hough para a Detecção de Traços num Detector de Fibras Óticas Cintilantes”. In: *XXI Encontro Nacional de Física de Partículas e Campos, São Lourenço.*, 2000.
- [77] KOSTIN, *Implementação Paralela da Transformada de Hough para Detecção de Traços em Física Experimental*. Universidade Federal do Rio de Janeiro, 1999.

- [78] F. FRANÇA, JOSE MANOEL DE SEIXAS, A. C. H. D., “Parallel Implementation of a Track Recognition System Using Hough Transform”. In: *Lecture Notes in Computer Science, Alemanha, p. 467-480*, 2001.
- [79] GROUP, A. D., “ATLAS Data Challenge 1”. In: *Geneva, CERN*, April 2003.
- [80] CHRISTENSEN, C., “Pythia Documentation”, <http://atlas.web.cern.ch/Atlas/GROUPS/SOFTWARE/OO/dist/7.0.0/InstallArea/doc/Pythia/html/index.html>, Agosto 2005.
- [81] LUEHRING, F., “Athena Pile-up Requirements”. In: *version 2.1, Geneva, CERN*, September, 2001.
- [82] COMMUNITY, A. H. L. T., *High Level Trigger Technical Design Report*. ATLAS CERN, May 2003.
- [83] GOLDFARB, S., (EDITORS), A. S., “Definition of Offline Readout Identifiers for the ATLAS Detector”. In: *ATLAS-SOFT-2001-004*, 2001.
- [84] N. KONSTANTINIDIS, H. D., “Determination of the z position of primary interactions in ATLAS”. In: *Nota técnica do ATLAS no. ATLAS-SOFT-2002-007*, 2002.
- [85] N. KONSTANTINIDIS, H. D., “Algorithms to select space points of tracks from single primary interactions in ATLAS”. In: *Nota técnica do ATLAS no. ATLAS-COM-DAQ-003-040*, 2003.
- [86] *ADSP-21160 EZ-KIT Lite, Evaluation System Manual*. Analog Devices, Inc., March 2004.

# Apêndice A

## Publicações

Este apêndice apresenta os artigos publicados em congressos nacionais, internacionais e notas técnicas sobre o trabalho realizado.

- (1) APLICAÇÃO DA TRANSFORMADA DE HOUGH PARA LOCALIZAÇÃO DE RETAS EM IMAGENS VOLUMÉTRICAS COM ELEVADO RUÍDO DE FUNDO, A. Gesualdi, J. M. Seixas, M. P. Albuquerque e G. M. Hospodar, submetido à revista SBA, abril, 2006, português.

RESUMO: Este artigo apresenta a aplicação da transformada de Hough para detecção de retas em imagens volumétricas do detector de ATLAS. Este detector busca por novas descobertas na área de física de partículas, através da colisão de prótons a altas energias. A detecção precisa do ponto de colisão possui uma importância fundamental para reconstrução de partículas, que consiste em um dos maiores desafios do ATLAS. O ponto de colisão é obtido através da detecção de retas na imagem volumétrica que representa o produto das colisões. Os algoritmos de localização deste ponto possui uma latência de execução de 1 ms. Portanto, as técnicas de detecção devem ser rápidas e precisas. A técnica de transformada de Hough nas formas local e adaptativa foram aplicadas com sucesso a este problema, atingindo uma eficiência de detecção de 89% em aproximadamente 0,32 ms. Este artigo apresenta os algoritmos de Hough utilizados, bem como o estudo desenvolvido da seleção de pontos da imagem volumétrica.

ABSTRACT: This paper describes an application of Hough transform to detect lines in volumetrical images from ATLAS particle detector. The ATLAS detector will search for new discoveries in the head on collisions of protons of extraordinarily high energy. An accurate detection of the collision point has a fundamental importance for particle reconstruction, one of ATLAS biggest challenge. The collision point is obtained by detecting lines in the volumetrical image produced by the debris of the collisions, and have an execution latency of 1 ms. Thus the detection technique has to be precise and fast. The Hough transform technique in its local and adaptive form were successfully applied to this problem, achieving a detection efficiency of 89% in approximately 0.32 ms. This paper presents the Hough transform algorithms used and a study of selection of relevant points from the volumetrical image.

- (2) LOCALIZAÇÃO DE VÉRTICES PRIMÁRIOS DE COLISÃO DO ATLAS POR TRANSFORMADA DE HOUGH, A.G. Mello, J. M. Seixas, G.M. Hospodar, M.P. Albuquerque, ENFPC 2005, português.

RESUMO: Nos modernos experimentos com colisionadores de partículas de altas energias, os detectores de traço estão sempre presentes. Estes detectores possuem a tarefa de localizar o vértice primário de colisão e reconstruir as trajetórias descritas pelas partículas. Este artigo aborda a construção de um sistema *online* para o processamento de imagens de colisões de partículas. Um evento de colisão é representado por uma imagem volumétrica de 7 camadas, com granularidade diferente por camada. O desafio consiste em utilizar o conceito de seleção de regiões e a transformada de Hough para extrapolar a informação de trajetória contida nestas imagens, a fim de localizar o ponto de colisão, ou vértice primário. A imagem volumétrica do evento é proveniente dos detectores de traço do experimento ATLAS no CERN. Neste sentido, o foco do estudo consiste na localização de vértices primários de elétrons (elétrons com empilhamento) com elevada acurácia e mínimo tempo de execução.

No contexto do experimento ATLAS, a geração da imagem volumétrica dá-se através do sistema de filtragem TDAQ (*Trigger and Data Acquisition*). Estima-se que o volume de dados inicial do sistema TDAQ seja de 160 Gbyte/s, e que

a massa de dados filtrada seja de aproximadamente 1,5 Mbytes/colisão, ou algumas centenas de Mbytes/s. De modo a operar com estas especificações, o sistema TDAQ possui três níveis de filtragem, que são executados em cascata, com complexidade e tempo de processamento crescente. Este trabalho está sendo realizado no segundo nível de filtragem, aplicando a transformada de Hough, em sua forma adaptativa, para a localização do vértice de colisão. Os resultados ilustram que esta transformada soluciona casos de radiações bremsstrahlung que não são contempladas pelos sistemas atuais, se adequando a restrição de operar com latência de execução inferior a 10 ms. Este trabalho está sendo realizado em colaboração com o grupo PESA (*Physics and Event Selection Architecture*) do ATLAS, utilizando o ambiente de desenvolvimento ATHENA. A transformada de Hough adaptativa atinge eficiências de  $\approx 86\%$  para eventos de elétrons com alta luminosidade e com empilhamento. Os tempos de execução da abordagem por Hough adaptativo mostram que são compatíveis com a latência de execução, possuindo a média de 2ms nas máquinas LXPLUS do CERN, operando o sistema ATHENA 10.1.0.

- (3) ALGORITHMS FOR THE ATLAS HIGH-LEVEL TRIGGER, S. Armstrong, J.T. Baines, et al, IEEE Transactions on Nuclear Science, vol. 51, no. 3, junho 2004, inglês.

RESUMO: Este artigo apresenta a arquitetura de orientação a objetos desenvolvida pelo segundo e terceiro níveis de filtragem do ATLAS. Os componentes funcionais dentro do sistema que são responsáveis para gerar os elementos de decisão de filtragem são algoritmos dentro de uma arquitetura computacional. Os aspectos relevantes da arquitetura são revisados em exemplos concretos de alguns algoritmos específicos e suas eficiências nas seções “verticais” de várias estratégias de seleção em física.

ABSTRACT: Following rigorous software design and analysis methods, an object-based architecture has been developed to derive the second- and third-level trigger decisions for the future ATLAS detector at LHC. The functional components within this system responsible for generating elements of trigger

decisions are algorithms running within the software architecture. Relevant aspects of the architecture are reviewed along with concrete examples of specific algorithms and their performance in “vertical” slices of various physics selection strategies.

- (4) AN OVERVIEW OF ALGORITHMS FOR THE ATLAS HIGH LEVEL TRIGGER, S. Armstrong, J.T. Baines, et al, ATLAS internal note, ATLAS-CONF-2003-003, junho, 2003, inglês.

RESUMO: Seguindo os métodos rigorosos de projeto de software e análise, uma arquitetura baseada em objetos foi desenvolvida para realizar decisões de filtragem dos segundo e terceiro níveis do detector ATLAS no LHC. Os componentes funcionais dentro do sistema que são responsáveis por gerar elementos de decisão de filtragem são algoritmos dentro de uma arquitetura de software. Os aspectos relevantes da arquitetura são revistos dentro de exemplos concretos de algoritmos específicos.

ABSTRACT: Following rigorous software design and analysis methods, an object-based architecture has been developed to derive the second- and third-level trigger decisions for the future ATLAS detector at the LHC. The functional components within this system is responsible for generating elements of the trigger decisions are algorithms running within the software architecture. Relevant aspects of the architecture are reviewed along with concrete examples of specific algorithms.

- (5) AN IMPLEMENTATION OF REGION-OF-INTEREST SELECTION FOR ATLAS HIGH LEVEL TRIGGER ANF OFFLINE SOFTWARE ENVIRONMENTS, A.G. Mello, S. Armstrong, S. Brandt, ATLAS internal note, ATLAS-COM-DAQ-2003-005, março, 2003, inglês.